

# To CRYPTOGRAPHY

- What is cryptography - an introduction
- Secret communication - a coded history
- Simple ciphers explained
- Uses of cryptography
- Cryptanalysis and decryption
- The lore of unbreakable ciphers
- DIY - Make your own cipher
- Future: Quantum cryptography





**www.  
thinkdigit/forum**

Join the forum to  
express your views  
and resolve your  
differences in a more  
civilised way.

**thinkdigit  
FORUM**

Post your queries  
and get instant  
answers to all  
your technology  
related questions



One of the most active online technology forums  
not only in India but world-wide

**JOIN  
NOW**



**www.thinkdigit.com**



# CRYPTOGRAPHY

---

powered by



# CHAPTERS

## CRYPTOGRAPHY

NOVEMBER 2012

06

PAGE

### What is cryptography?

This chapter gives you an introduction to the mysterious world of ciphers, codes and other secret forms of communication

17

PAGE

### Secret communication: A coded history

Since the dawn of time mankind has been developing secret codes. This chapter follows this evolution

36

PAGE

### Simple ciphers explained

Starting off with the basics, we take you through the elementary types of ciphers out there

50

PAGE

### Uses of cryptography

Let's now discuss the expanded role and usefulness of Cryptography in modern times

## CREDITS

The people behind this book

### EDITORIAL

#### Executive Editor

Robert Sovereign-Smith

#### Writers

Abhishek Choudhary  
Priyanka Mathur  
Shashwat Shukla  
Spandan Sharma  
Tuba Raqshan  
Vaibhav Kaushal

### Features Editor

Siddharth Parwatay

### Contributor

Copy: Infancia Cardozo

### DESIGN

#### Sr. Creative Director

Jayan K Narayanan

#### Sr. Art Director

Anil VK

### Associate Art Directors

Atul Deshmukh  
Anil T

### Sr. Visualisers

Manav Sachdev  
Shokeen Saifi

### Visualiser

Baiju NV

60  
PAGE

## Cryptanalysis and decryption

We have been breaking codes as long as they have existed. This chapter looks at the science behind code-breaking

66  
PAGE

## The lore of unbreakable ciphers

From time immemorial, mankind has been obsessed with puzzles. The unquenchable thirst to find answers has become the defining characteristic of our species.

79  
PAGE

## DIY: Make your own cipher

So now that you know a decent bit about cryptography, how about making your own little cipher?

88  
PAGE

## The Future of ciphers

Change is the only constant, because nothing else can afford to stop

### © 9.9 Mediaworx Pvt. Ltd.

Published by 9.9 Mediaworx

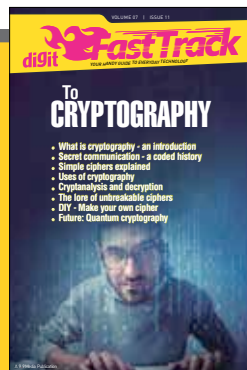
No part of this book may be reproduced, stored, or transmitted in any form or by any means without the prior written permission of the publisher.

### November 2012

Free with Digit. If you have paid to buy this Fast Track from any source other than 9.9 Mediaworx Pvt. Ltd., please write to [editor@thinkdigit.com](mailto:editor@thinkdigit.com) with details

### Custom publishing

If you want us to create a customised Fast Track for you in order to demystify technology for your community, employees or students contact [editor@thinkdigit.com](mailto:editor@thinkdigit.com)



COVER DESIGN:  
PRAMEESH PURUSHOTHAMAN

# Introduction

Everyone has secrets. And as ironic as it sounds, what good is a secret if you can't share it with someone? But the operative word here is someone, and only that someone. How do you do that? You devise a secret language or code that can only be understood by those in the know. Even as kids, we all developed our own little "code words" as ways to outsmart adults, or even other kids, and create some sort of perceived privacy from prying ears.


Cryptography is this very science of secret communication. The science dates back to the very dawn of civilisation, and over the years has had a significant part to play in everything from war, literature, culture, technology and to even crime. For years cryptography had an aura around it of being an occult practice, understood and used by a select few - shadowy characters who operated in secret and did things you'd be better off not knowing. And then Dan Brown's *The DaVinci Code*, changed things. The layman was introduced to the concept of cryptography and soon codes in plain sight were noticed and understood by all, because you had the right "key" to decipher them. This Fast Track aims to go a step further. By the end of it, you will be able to make your own ciphers and codes. How? By building a sound foundation using the various concepts of cryptography. Transposition ciphers, substitution ciphers and other basic techniques may be a far cry from the sophistication of ciphers in use today to protect the vast oceans of digital information we now create and transmit, but these are the building blocks on which rests this fascinating field of cryptography.

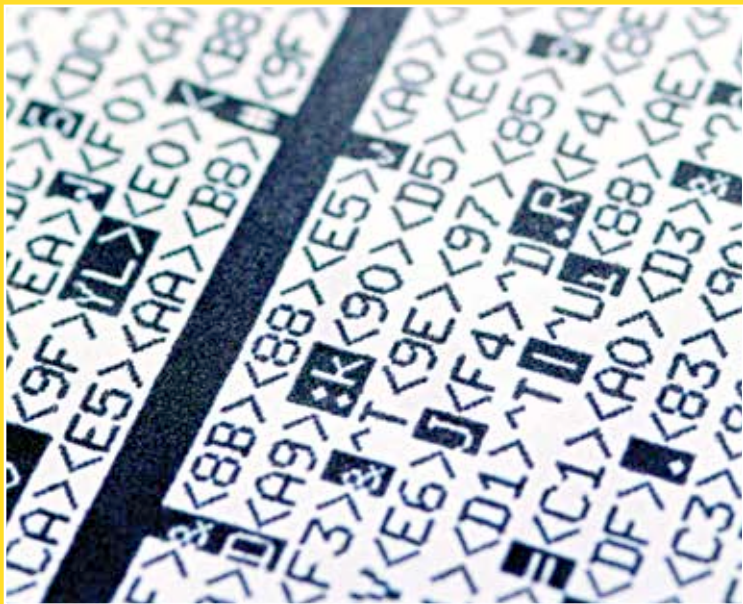
The book starts off with these basics, explaining key terms and other jargon associated with this science. It then takes you on a historical journey peppered with anecdotes of the role cryptography has played in toppling governments, winning wars and changing history as we know it. From there you are put in the driver's seat. Imagine you discovered an encrypted message - a sheet of seemingly garbled gibberish - your mission (should you accept it) is to decrypt this message. It's a matter of national security! How would

you do it? Using Cryptanalysis - the technique of decrypting coded messages.

By now some of you, especially those of you who have played our Crack The Code game, must be convinced that any code created can be cracked. Not true. Enter the one-time pad cipher, and other enigmas such as Kryptos that have haunted mankind for years. Want to take a crack at those? Who knows, maybe you're the genius the world has been waiting for.

We round off the book with a short section on making your own cipher (as promised), and an introduction to how the invention of quantum computing is going to drastically change the face of modern cryptography as we know it.

Although this FastTrack will get technical whenever needed, it's not meant to present concepts at the level of a Ph.D thesis. Those mathematically inclined should be inspired to take up the subject seriously. This is one place where we don't mind if this FastTrack serves as your "gateway drug" into this mysterious world. Happy reading! Or should we say, Ibqqz sfbejoh! 



# WHAT IS CRYPTOGRAPHY?

This chapter gives you an introduction to the mysterious world ciphers, codes and other secret forms of communication

**E** PFT UIJT MPPL MJLF OPOTFOTF UP ZPV? Does this look like nonsense to you? Did you notice that we asked you the same question twice? Confused?

Look at the set of random alphabets in that first “sentence”. Replace each letter with the letter which comes before it in the alphabet e.g. replace B with A, P with O, F with E and so on and reread the sentence. It’s easy but in case you’re slow (which we doubt considering you’re a Digit

reader), let us help. That would translate to: “DOES THIS LOOK LIKE NONSENSE TO YOU?”. Now read the third sentence and the confusion should go away. You see, cryptography is simple! All it takes is a trick up your sleeve and the patience to decode or encrypt a message.

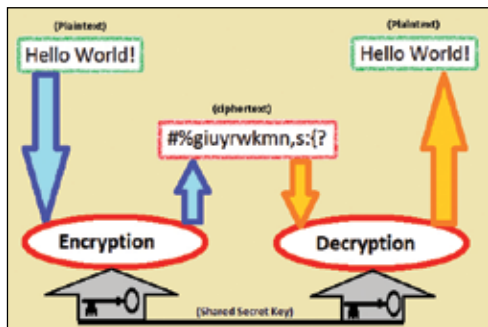
## What is Cryptography?

Cryptography is the art and science of converting ordinary information into gibberish and converting it back to its original, meaningful form. It can be done using either simple procedures or very complicated mathematical algorithms. Don't be intimidated by its complexities though. As always, we're here to simplify it for you. And don't forget, you did manage to understand the first code we threw at you, didn't you?

It's amazing how rotating the alphabets by even just one position is enough to confuse the human mind! Oh, and we call it 'rotating', not 'shifting' because shifting would mean having 'Z' replace 'A' as well.

Cryptography involves tricks like there to scramble messages to hide them from unintended audiences. The above example could be easily understood after basic trial and error even if the code breaker didn't have the instructions required to unscramble it. To make it unbreakable, we'll need to employ a sophisticated algorithm. But let's get to that later when you'll be familiarized with some simple algorithms. You'll also learn to create your own small algorithm to encrypt and decrypt data and it's going to be fun.

One of the most prominent secret language systems in India is the famous tapori language well known by Mumbaites. With keywords such as peti, khokha and supari, the language has found wide acceptance in Bollywood movies, especially those revolving around organized crime. Though these words are famous now, they were once known only by street thugs. Notice that the vocabulary consists of actual words from the Hindi language whose meanings differ when used by the seasoned tapori. This is a good example of cryptog-



Symmetric Cryptography

raphy gone wrong – not due to the nature of its use but due to its failure to remain a secret language.

## Who uses Cryptography?

You. Yes, you do use cryptography. How? Answer the following questions:

1. Do you use Facebook?
2. Do you use Gmail?
3. Have you ever opened a PDF or ZIP file that asked you a password?
4. Did you ever use an *HTTPS* site?
5. Do you chat with your friends on the internet?
6. Do you fill forms on the internet?

If the answer to all the questions is ‘no’, then maybe you don’t use cryptography. Our guess is there’s a very rare chance that you answered with a ‘no’ to all the questions. Cryptography is everywhere and we use it daily. It doesn’t only involve using arcane commands on a black terminal. It’s present in the day-to-day tools we use – zipping software, PDF viewers, word processors, browsers and what not. People who don’t use computers also use cryptography.

## Day-to-day tricks we use

You’ve probably heard of celebrities changing contact names in their phone books to avoid having to admit to infidelity. You must have done it too but for more worthy reasons such as for keeping a relationship under

wraps by changing your boyfriend’s contact name in your address book to ‘Vampire’ just because he is as loving to you as Edward was to Bella, then you have used cryptography. You may however need to get a life but that apart, changing your sweetheart’s name to one belonging to the opposite sex is nothing but cryptography.



Twitter over HTTPS. Most web browsers allow you to see the cryptographic information of a secure page by clicking the favicon on the address bar

## What is not Cryptography?

Let’s say you skipped work for a day to party with some friends. You

have only your office computer on which you can save the pictures taken that day. Since you can't keep them in the main photo collection, you put them inside the C:\Windows\System32\Config folder. That's clever since no one will ever look for pictures in that folder. Did you just do some crypto? Well, the answer is no.

Cryptography is the science of scrambling information. Hiding information cannot be called as cryptography. Hence, uploading files to your Dropbox or Google Drive account whose password is unknown to anyone but you would still not qualify as cryptography.

It does, however, if you zipped the set of pictures into a zip file with a password. This is because the software you used for zipping with a password (e.g. Windows shell or WinZip) actually uses an algorithm called AES to encrypt the data – in other words, it modifies the data.

You might ask “Even zipping pictures into a zip file without a password would modify the data. Why doesn't that count as cryptography?” Remember that cryptography doesn't just involve ‘altering/modifying the data’. It is a method to ‘alter the data in such a way that unintended people are unable to interpret it’. Given the easy availability of compression software which can easily detect a ZIP file (even if you change the extension), just compressing the picture set won't protect it.

Another question might arise at this point. At the beginning of this chapter, we used a technique of rotating the letter of the alphabet – is that a part of cryptography given that it was even easier to crack than it would be by compressing/decompressing it? Well, it depends. Altering information to render it incomprehensible is one part of cryptography. If the trial and error method to understand the message works and you didn't need to know that rotating the letters of the alphabet was the key, then it's no cryptography at all. But to someone who can't figure out what those random letters mean, the message remains unbreakable and thus, successful confidentiality is achieved. Once again though it depends, on who's attempting to break the code. Given the intelligence of our species which aims to colonize Mars, it wouldn't be an unimaginable feat for people of a certain level of intelligence. In such a case, the message encrypted using the ‘alphabet rotation’ technique is vulnerable to being intercepted and would qualify as a weak cryptographic method.

## Breaking down the jargon

If you've been using computers since some time and have read this far, you might wonder who really uses these techniques in real life. You're right,

not many people do. If you wanted to secure a name in your phonebook, you'd probably use an app on your smartphone or an online web service to do the job. It is this very laziness that contributes to cryptography's effectiveness. The fewer the people who know about it (or are bothered to learn about it), the more successful you are at communicating messages without interception.

When you proceed towards learning about cryptography, you'll come across many terms. Terms such as keys, ciphers, encryption, codes, algorithm etc. Here we'll talk about them briefly so that you won't be intimidated by them. Also, you'll come across the jargon time and again while reading this little book so it's better to know them beforehand. Let's start with the simplest one.

## **Plaintext**

All scrambled messages were once pieces of texts. These pieces which are yet to be scrambled are called 'plaintext' in cryptography vocabulary. Back when the science of scrambling messages started taking shape, it was usually text that needed to be sent securely. It's due to this reason that the original message which needs to be encrypted is called plaintext.

In the present context though, the word 'plaintext' doesn't justify all cases of encryption. We strive to keep all data communication secure and that surely doesn't include just text. Images, videos, programs and documents that don't fit into the 'simple text' mold are also transferred securely. Regardless of all this, you'll find numerous instances of data that needs to be encrypted being called plaintext.

## **Ciphertext**

After you have scrambled the plaintext, you get an output which should pretty much look like mystical nonsense. This nonsense is called 'ciphertext'. It's also noteworthy that the algorithms to encrypt data are often called as 'ciphers'. Thus, the output they produce is called ciphertext.

## **Encoding & Decoding/ Encryption & Decryption**

There's no shortage of examples where 'encryption' and 'decryption' are commonly mistaken for 'encoding' and 'decoding'. The reason for this is the similarities they share. Both transform data into another format and both are reversible (unlike 'hashing'). Though there are differences (Flip over to Chapter 3 for indepth information on the differences), all four are a part

of cryptographic literature. For the record, the alphabet rotation technique in the beginning is an example of encoding/ decoding.

To avoid confusion, for now let's assume that the terms are interchangeable.

## **(Cryptographic) Algorithm**

In computer science, algorithms are a set (series) of instructions which when executed would solve a problem. The word retains its meaning in cryptography. The algorithm one uses to encrypt / decrypt messages is called 'cryptographic algorithm'. Note that when we talk about cryptographic algorithms, there will always be a pair. The first algorithm of the pair is responsible for encryption of the data i.e. to convert plaintext to ciphertext and the second one is responsible for decryption i.e. to convert the ciphertext back to plaintext. Though not in all, but in many cases (typically symmetric algorithms) the decryption algorithm is basically the reverse of the encryption algorithm and vice versa. For this reason, in most literary works on the science of cryptography, both the algorithms are referred to as one algorithm only. They're usually also implemented as a single program, hence the style of reference.

## **Encryption Key**

Encryption or ciphering requires two things – an algorithm and at least one key. The algorithm converts the plaintext into ciphertext based on both, the algorithm as well as the key. If you change any one of them, the result will differ. Taking again the alphabet rotation example, the algorithm is 'Forward Rotation' and the 'Key' is '1'. If we would have rotated the alphabet by 2 places (e.g. replace A with C, B with D and so on), then the key would have been '2'.

Note that if you change the algorithm (like from forward rotation to backward rotation) or the key (e.g. rotate 3 places or 7 places) then the ciphertext will change. A good algorithm and a strong key are the necessary properties to make an encryption system strong.

## **Symmetric Ciphers**

'Symmetric ciphers' or 'symmetric algorithms' are algorithms which use the same key to encrypt as well as decrypt data. Traditionally, this type of algorithm has been in use in methods of cryptography. The example that we gave is one of a symmetric cipher. When you encrypt the plaintext by replacing each letter with a letter one place ahead in the alphabet then you

can't revert to the plaintext by replacing the letters in the ciphertext by letters two places back. You'll have to replace the letters by the same number of positions backwards.

Symmetric ciphers are named as such because both sides of the process (encryption and decryption) utilize the same key. Symmetric ciphers are also known as 'private key cryptography'.

## Asymmetric Ciphers

Asymmetric ciphers are usually known as public key cryptography and are well known as 'public key cryptography' algorithms. In a public key cryptography method, the key which is used to encrypt plaintext to ciphertext can't be used to decrypt the ciphertext back into plaintext. The two keys involved are called as public key and private key.

**Public Key:** Public keys are used to encrypt plaintext. They're named as public keys because they can be made public.

**Private Key:** Private keys are used to decrypt ciphertext back to plaintext. They're named as such because they're meant to be private.

Both these keys are related to each other using a mathematical relationship in a way that if you're given one key, you wouldn't be able to derive the second key. This allows the public key to be public. Anyone can have it and send you a message encrypted with the public key but no one can decrypt an encrypted message being sent to you by another person.

Let's consider a hypothetical situation. Agent Archer wants to send a secret message to his boss, Jim; so he asks Jim for his public key. Jim creates a public-private key pair and sends the public key to Archer. A spy named Cyril uncovers the public key! Agent Archer then encrypts the message and sends it to his boss. The spy also intercepts the encrypted message. In the meanwhile, Jim has also received the encrypted message and he uses his private key to decrypt the message. Cyril is still trying to figure out how to decrypt the message using the key!

This property of asymmetric algorithms makes them indispensable in the modern era. They allow to freely send the key as well as the message! Since the key for decryption is separate from what is being used for encryption, it makes the communication safe!

However asymmetric algorithms are usually much more costly on the CPU than symmetric algorithms. This brings up an interesting usage of public key algorithm (asymmetric algorithm) whereby it is only used to transfer the key of a symmetric algorithm. This allows the key to be trans-

ferred securely while minimizing the amount of computational power required to do the actual encryption or decryption.

## Key strength

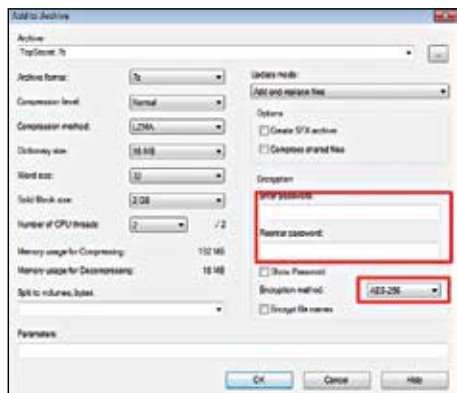
Key strength usually refers to the length of the key being used for the algorithm. However it isn't always the measure of its cryptographic strength. Randomness of characters used in the key is also an important factor in determining the strength of the cipher.

## Hashing

While you can encrypt passwords for any system (website user passwords, operating systems account passwords or passwords to access a program), it's possible to decrypt them. Since passwords are still the major way to authenticate users for various purposes, they do pose a threat to security.

Make that a serious threat to security in the case of Open Source systems such as Linux. It's common knowledge that Open Source can allow hackers to know about the weaknesses or loopholes of the software more easily than Closed Source software. If the password file gets into the wrong hand, security would be a thing of the past. Obviously, storing passwords in plaintext (original) format would be downright stupid!

For these reasons passwords are neither stored in encrypted form nor in plain format. So how are they stored? Well, they're stored in 'hashed' form. Hashing is a technique to map text of variable length to a string of fixed length in such a way that the only way to find out the original text from the hash is to run a brute force attack. Since hashing functions are irreversible, they don't qualify strictly as cryptographic functions. However, they're associated with security often enough and it's easy to find them mentioned alongside passwords in the context of cryptography.



Compression software such as 7-zip help in encrypting compressed archives

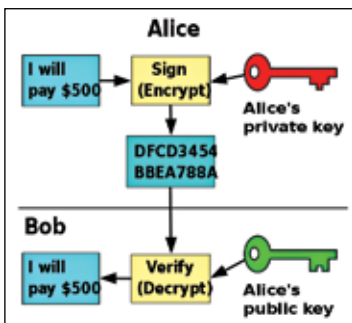
## Cryptographic algorithms and security – married by keys

Though we have a chapter dedicated to cryptanalysis, we'll give you quick rundown on what it takes to break a cipher and what factors govern the security of a cryptographic algorithm.

### Algorithm strength

A very strong key and a weak algorithm will result in a weak cipher. For convenience sake, let's take the example we showed in the beginning again. We're rotating the alphabet by  $n$  number of times. Here  $n$  is the key. Now, it won't matter if we put  $n = 200$  or  $n = 1000$ , there's a limit to its effectiveness. The effectiveness limit is 25 times. You can't rotate alphabets by more than 25 times. The 26th time it would always come back to the same position as if the rotation was never done. In simple terms, the algorithm itself is limiting the difficulty it can pose to an unintended recipient against cracking the cipher (converting ciphertext back to plaintext).

Not only this, the algorithm is very simple too. Rotating alphabets is something a child can do easily to pass messages to his friends in a classroom. It's no big deal. Consider a scenario where the teacher catches the paper on which he was passing the message. The message is encrypted but how difficult would it be for the teacher to interpret? All she would have to do is put herself in the child's place thinking "what can a child do to encrypt a message?" Add some trial and error into the mix and the message being passed would be easily interpreted by the unintended recipient (the teacher in this case). Hence, for maintaining security, using a good algorithm is important.



Asymmetric algorithms are also used for digital signatures

### Computational power required

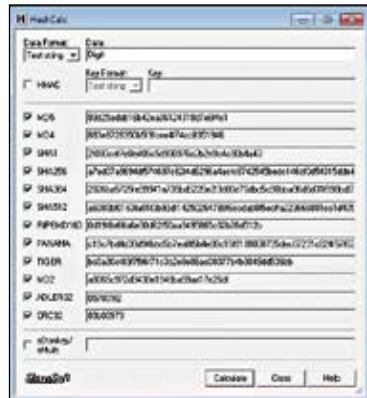
Okay, so you own a mobile phone that's more powerful than a computer NASA sent to the moon years ago. But is that power enough to break a DES cipher? What about a Triple DES? Or RSA perhaps? Nope, right? Your mobile phone wouldn't be able to do it. If you have the ciphertext and the algorithm that was used, then you need to know

the key in order to break the cipher. In most cases, the only way to figure out the key would be to run a brute-force attack which would use the algorithm with every possible input as a key. Once the data is decrypted in an acceptable pattern, it can stop and record (and display) the key that was used for encryption. Now, when it comes to brute-forcing a cryptographic algorithm, it's difficult to do it for a simple reason - most algorithms are computationally intensive and require more computational power. To break them would take plenty of computational resources and time. That brings us to another factor - time.

## Time

We mentioned that your mobile phone can't crack DES or RSA algorithms, but it in fact can crack any algorithm. The only problem - it will perhaps take millions of years to break it! That would be pretty impractical, wouldn't it?

Apart from requiring a strong algorithm and huge amounts of computational powers, it would take time to break a cipher. However we have an interesting case - DES. DES was an algorithm which was developed in the 1970s and was so popular that it became a standard. The algorithm's full name happens to be 'Data Encryption Standard'. When it was invented, it was said to be very strong. Computers in those days would have taken eons to break it. Today, many loopholes have been discovered in the algorithm and it has been proven vulnerable. Time is an interesting factor when it comes to cryptography however, in most cases, the data protected by an algorithm which is deemed to be strong in its own time becomes useless by the time the cipher becomes vulnerable due to further research in the area or due to increase in computing capability.




HashCalc is a good tool with small footprint to generate hashes for popular algorithms

## Encryption keys

RSA is widely known for being a difficult nut to crack! This is because RSA is an algorithm that depends on the difficulty of finding the factors of the product of two prime numbers. The product of two prime numbers

is an important part of the key in RSA algorithm (RSA is an asymmetric algorithm). The truth, though, is that RSA is very much crackable. If you want to find out the factors of a number below a million which is a product of two prime numbers then modern computers would probably be able to do that in a few seconds! And yet, RSA is used widely. Why?

The reason is that an RSA's security increases as you use a larger number for a key i.e. if you choose a number which is a product of two very large prime numbers then RSA becomes very difficult to crack. The use of this algorithm exhibits the importance of strong keys for gaining more security! 



# SECRET COMMUNICATION – A CODED HISTORY

Since the dawn of time mankind has  
been developing secret codes. This  
chapter follows this evolution

**T**he year 2003. The world woke up with a new perspective on history, a revelation that would shock many and create avid interest in the art of secret communication. Dan Brown's *Da Vinci Code* created a furore among critics and readers alike. While the former chose to point out the literary inconsistencies, the latter rushed to read up more on the hidden layers of Da Vinci's art – in other words, their first ever tryst with cryptography. However, this science of secret communication has been as old as mankind itself. Well, almost.

## Starting simple

Early evidence of cryptography indicates that it has been around since Biblical times. The Bible Code, also known as the Torah Code, is supposedly a series of secret messages embedded in the text of the Torah. This has been decoded using ELS (Equidistance Letter Sequence) and is a highly debatable topic.

It is believed that cryptography began approximately around 2000 B.C. in Egypt. The famous hieroglyphics adorning the tombs of rulers depicted the lives of the kings and the great acts done by them. Intentionally kept in a cryptic mode, these hieroglyphics were only used for an ornamental purpose rather than for secrecy. Soon, this system became obsolete.

## Storing Intelligence

Since cryptography is mysterious in its approach, this science began to be associated with the dark arts. And since most early cryptographers were usually the intelligentsia or the scientists, commoners considered them to



Some say secret codes and symbolism exist in Da Vinci's paintings

be the devil's followers. In the Middle Ages, many writers, who were also members of secret political or religious organizations, found an innovative way to communicate through codes. They published books which contained ciphers. This led to a trend of secret writing and every royal court across Europe competed to encrypt ciphers that couldn't be broken. During this period, many intellectuals chose to hide their scientific inventions and discoveries in secret code for the fear of being excommunicated by the Church. However, these mavericks trusted the future generations to unveil their discoveries, understand the implications and appreciate their efforts.



Masonic symbols

## Early ciphers

The earliest and most commonly used cipher was the Atbash cipher, which emerged from Mesopotamia. Sharing many similarities with its Egyptian counterpart, cuneiform script was used to decode the message. Used in Babylon and Assyria, the Atbash cipher substitutes the last letter of each alphabet by the first, as seen below:

ABCDEFGHIJKLMNOPQRSTUVWXYZ  
ZYXWVUTSRQPONMLKJIHGFEDCBA

The earliest cipher dealt with symbolism and is a key to an immense source of ancient wisdom, both scientific as well as philosophical. Early philosophers chose to subtly encode their message in verbose literature, intentionally meant to be heavy-worded. These ciphers were concealed artfully, either in a watermark, or through a repeated sequence appearing in a particular order. Under the guise of subtlety, many a carefully hidden message made its way to the recipient it was intended for.

## Freemasons and their secrets

The highly controversial organization of Freemasons has brought to light the many uses of ciphers. While mostly misunderstood, Freemasons were essentially a group of highly intellectual individuals, chiefly believing in science. Their immense store of scientific knowledge was kept hidden through complex symbols and ciphers, sometimes even for years. Usually,

these symbols were imbibed from nature with a deep reverence, which only those initiated into the ways of the organization could comprehend.

It was during the middle ages that ciphers became increasingly popular. Governments used them to communicate with their ambassadors in other countries, and it was not only the Western nations which had begun to understand the importance of having an unbreakable code.

## Eastern codes

The Ancient Chinese utilised the pictorial (ideographs, to be precise) form of their language to enclose the deeper meanings attached to the words. Most often, messages would be conveyed in ideographs to maintain secrecy. However, such use of cryptography is not apparent in any of China's earlier military conquests.

History claims that most ciphers in India were highly advanced, with the Government employing complicated codes to communicate with their spies distributed across the length and breadth of the country.

However, one of the first credible advances in this field was devised in Italy. An exhaustive organization was set up in Venice in 1452 to deal with cryptography. This ensured that all ciphers were solved and segregated for the Government's discretion.



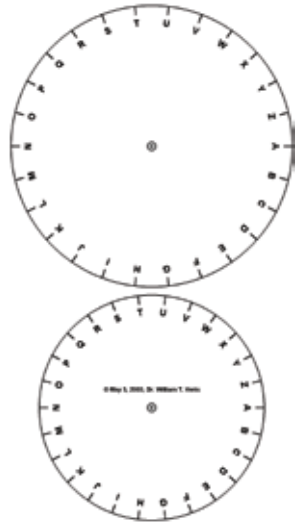
First mention of cryptography in Iliad

## All In Greek

The Greeks are believed to have been well acquainted with the use of ciphers, which were right before the eyes but seldom noticed. The Greek historian Herodotus revealed instances of secret messages hidden beneath the wax on wooden tablets and also of situations where discrete information was tattooed on a slave's head and covered by regrown hair. Those were perhaps the crudest forms of steg-

anography, a form of cryptography. Steganography is a form of security through obscurity, or in simple terms, secrets hidden in plain sight. It not just conceals the message but also protects the persons communicating it.

Snatches of cryptography are believed to have been used in the popular Greek epic, the Iliad, especially when Bellerophon was presented before the king with a secret tablet. This tablet informed the king that Bellerophon must be executed. The king tried doing so by having him fight many mythical creatures. But to his chagrin, Bellerophon won every battle.



A caesar cipher

## Binary Inspirations

Spartans employed a method called the Skytale (or Scytale) cipher, where a thin sheet of papyrus was wrapped around a staff. Then, the messages were written down all along the staff. The papyrus was then unwrapped. To decode and read the message, the papyrus would need to be wrapped around a staff of an equal diameter. Greek warriors used this technique to communicate. At that time, it was quite effective as only the right staff would yield the message. The Spartan military was believed to have used the Scytale transposition cipher extensively.

Yet another Greek discovered an ingenious cipher. Polybius used a cipher where the letters of the alphabets were arranged in a five-by-five square (the Polybius Square). The letters “I” and “J” occupy the same square. It’s identical to the Playfair technique. The rows and columns are numbered 1 to 5, so every letter has a pair (either in a row or a column). The pairs can now be easily communicated by torches or hand signals. The Polybius Square came into being since it reduced the size of the symbol set. In a way, the Polybius Square can be considered as one of the earliest predecessors to the modern day binary codes.

## When In Rome

While the Greeks used cryptography for their military purposes, the Romans were not to be left behind. Julius Caesar discovered a unique way



"You decoded my cipher? Et tu Brutus?"

to communicate secretly. The Caesar Cipher has a letter shifted two places further in the alphabetical order. T shifts to V and Y moves over to A, to give an idea. Given below is the Caesar's code which was perhaps the first ever cipher to be used by children.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

CDEFGHIJKLMNOPQRSTUVWXYZAB

This is an example of a mono-alphabetic substitution cipher where each alphabet is assigned to another.

However, it was Leon Battista Alberti who laid the cornerstone to most modern cryptography. Also known as the Father of Western Cryptography, he is credited with the invention of the polyalphabetic substitution, also known as the Alberti cipher, which came about before Caesar's cipher. This technique enables different ciphertext symbols to portray the same plaintext symbol. A closer inspection reveals that this simple substitution makes it all the more difficult to interpret the code, especially using the method of frequency analysis.

Ingenuisly, Alberti designed this cipher after decoding others and understanding their vulnerabilities. This unique cipher was designed with two copper disks, fitting into each other. The disks have alphabets inscribed on them. A coded message can be sent by aligning a prearranged letter on the inner disk with the corresponding alphabet on the outer disk, which makes

up the first character of the ciphertext. A few words later, the disks are rotated and the new alignment of the letters will create a totally new cipher. By doing this, the effectiveness of frequency analysis becomes limited and the cipher emerges as a strong one.

Despite other limitations, the constant rotation of these disks would change the cipher countless times, within the message. At that time, this was a pioneering concept in cryptography.

## German influence

Trithemius, a German monk, gave cryptography a

major boost way back in 1518. Interestingly, his deep interest in all things occult drew him to the realm of secret communication. He authored six books on 'Polygraphia' and created a polyalphabetic cipher. It was a table consisting of repeated alphabets in each row, with a duplicate sequence above it having a letter shifted in the arrangement. The message can be coded when the first alphabet of the plaintext is substituted with the first row, the second letter from the second row and so on.

## English messages

The prevalence of cryptography in old England has been traced to the philosopher Roger Bacon and the poet Geoffrey Chaucer. During the reign of Henry VII, cryptography was used by supporters of Perkin Warbeck, who was the Duke of York and the younger son of Edward IV. Warbeck was a renowned figure in the English court. However, his attempts at annexing Cornwall failed and resulted in his execution.

Amidst the entire furore, a substitution cipher for Warbeck has been documented. The prosecution record of indictment of the Earl of Warbeck



King Ferdinand VII

recorded that while in prison, Warbeck delivered to one of his supporters a book called “ABC”, also called a “Crosse Rowe”. Each letter in the book had a corresponding sign written by Warbeck. This was to ensure that using this cipher, the supporter could write back a message which would not be understood by those who didn’t know the code.

## Diplomatic ciphers

Catherine of Aragon was believed to have learnt the art of cryptography and used to write in ciphers. This helped her play a vital role as a secret channel between King Ferdinand of Spain and King Henry VII of England. After King Henry VII’s death and her subsequent marriage to King Henry VIII, Catherine still didn’t relinquish her cipher to the ministers but handled the cipher for secret communication between the kings. In 1515, Catherine gave up her position as an ambassador when her worthy successor arrived from Spain.

Despite their advancements in other sectors, England remained largely hostile to the use of cryptography. Only near the end of Henry VII’s reign were ciphers used by the English court. However, ciphers were used by the Spanish ambassador, in the course of exploring diplomatic ties with England. While

the King of England chose to send his written plain text messages via sea, to protect their contents, King Ferdinand of Spain chose to communicate sensitive matters through ciphers with the help of Catherine.

Later on, ciphers were largely employed by English ambassadors to Spain to keep the government abreast of the situation in the host country. Ambassadors John Stile and Thomas Spinelly used ciphers to communicate with King Henry VIII.



King Charles 1628 AD

## A matter of code

After discovering the convenience and security which a coded message provided, most English kings began to write in cipher. Charles I was the monarch who made the most use of ciphers during his reign. During their separation, Charles I and Queen Henrietta Maria used complicated ciphers to correspond.

Charles I used many different ciphers during the period of 1640 to 1650 to communicate sensitive information. His recipients included Prince Rupert, the Parliament General and ministers in Oxford.



Marie Antoinette

## French connection

The highly controversial Queen of France and Navarre, Marie Antoinette relied on ciphers to communicate with Count Axel von Fersen after their flight to Varennes.

At the height of the French Revolution in 1791, King Louis XVI and Queen Marie Antoinette escaped France in disguise. But they were detected and detained at the town of Varennes and returned to Tuileries Palace. Count Axel von Fersen, a Swede, was an accomplice who worked disguised as a coachman in Paris. The next day, he got the information of their failure. Fersen then communicated with Marie Antoinette from Brussels in cipher. Marie wrote back indicating that she was safe and warned him not to return to Paris or try to communicate with her.

Between the years 1791 and 1792, more than sixty letters were passed between Fersen and Marie Antoinette, written using either cipher or an invisible, white ink. The cipher was a complicated one and followed the pattern of a polyalphabetic substitution cipher. The manuscripts found

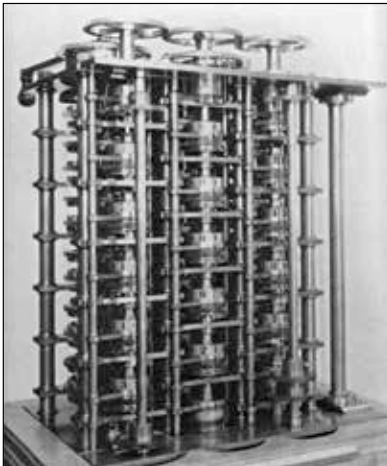
revealed that a keyword letter was aligned to every letter. Interestingly, since Fersen handled all the ciphered communication between the King and Queen during their escape, it is highly probable that Marie Antoinette also had the cipher.

### Read between the lines

Only later were the ciphered letters revealed as love letters by Marie Antoinette to Fersen. Apart from ciphers, they also used invisible ink, despite the problems it posed. It has been documented that Marie Antoinette found the invisible ink quite difficult to handle. And between them, they devised an ingenious way to communicate – lines of plaintext in invisible ink embedded between mindless texts in cipher, using the cipher as a guise.

### Modern Interpretation

The next major breakthrough emerged only much later, after the invention of the telegraph, which completely altered ancient modes of cryptography. Frenchman Blaise de Vigenere created a very practical method for a polyalphabetic system, named after him. The Vigenere cipher is a simple polyalphabetic substitution method which uses a series of various Caesar ciphers to encrypt a message. This cipher has been reinterpreted many times and while it was easy to understand and put into practice, it was rendered unbreakable by beginners. Thus, it commonly came to be known as the ‘indecipherable cipher’.



Charles Babbage's Analytical Machine

### Code Wars

Despite the circuitous and mostly complicated history of cryptography, it wasn't until the 19th century that major advances were made. Cryptanalysis became quite popular and most European states had teams to work on breaking coded messages. Charles Babbage, the creator of the first prototype of the current computers, had done extensive work on mathematical cryptanalysis, especially of pol-

alphabetic ciphers. This was later redeveloped and published by Friedrich Kasiski, a Prussian cryptologist.

## Kerckhoff's Principle

During that time, cryptography was mainly a culmination of basic rules, developed and reiterated over time. Frenchman Auguste Kerckhoffs' essays on cryptography laid the cornerstone for military use of the science. These articles scrutinized the state-of-the-art military cryptography of the time and appealed for massive improvements in the French practice. Apart from practical advice and basic rules, it also had six principles of cipher design for the practical age.

These principles emphasized on the simplicity of the key (which could be memorized) and the belief that the cipher should be unbreakable and transmittable by telegraph. Furthermore, it also stated that the documents



Skytale

should be handled by a single person and the system should be easy, without any requirement of mental strain. However, it was his second principle that became quite popular and came to be known as Kerckhoffs' principle. It stated that, "The design of a system should not require secrecy and compromise of the system should not inconvenience the correspondents." This gave an impetus to cryptography and spurred further milestones.

The misuse of cryptography led to the execution of Mata Hari in the early 20th century. Similarly, during World War I, the German naval codes were broken by the Admiralty's Room 40 (British Naval Intelligence), which perhaps played an important role in detecting German activity in the North Sea, giving Britain an advantage in the battles of Dogger Bank and Jutland.

## Black Chamber

Cryptanalysis units sprouted across Europe and were known as the Black Chamber. It started off in 1628 when Frenchman Antoine Rossignol helped the French army by deciphering a captured message and subsequently, defeated the Huguenots. The victory ensured that Rossignol was called upon to decode ciphers for the Government. His method of deciphering the



Surrender of Cornwallis

message included two lists – one with the plain elements in the alphabetical order and code elements in a random arrangement, and the other to help with the decoding where the alphabets and numerals in plaintext were random and the code elements in order. After Rossignol's demise, his son and grandson continued his work. By then, there were many cryptographers in the employment of the French Government. They formed the Cabinet Noir (the Black Chamber).

In the 1700's "Black Chambers" were common in Europe. The largest one, however, was in Vienna, called the Geheime Kabinets-Kanzlei. It was led by Baron Ignaz de Koch and the organization was mainly used to sift through all the mail coming to foreign embassies, copy these letters, reseal and return them to the post office the same morning. The office also took charge of other political or military interceptions. There were almost a hundred letters read in a day.

The English Black Chamber came into existence due to the efforts of John Wallis in 1701. Before that, he had to solve ciphers for the Government from an unofficial position. When he died, his grandson, William Blencowe, coveted the position. Having been taught by his father, Blencowe was granted the title of Decypherer. In the cryptographic world, the English Black Chamber has the longest history of victories.

The Black Chambers of Europe solved many ciphers and were widely successful. But the period of peace tested their usefulness, which was largely miniscule. By 1850, these Black Chambers were dissolved.

## Early american ciphers

America had no centralized organization dedicated to cryptography. Here, decryption was undertaken by interested individuals. The first instance of encryption dates back to 1775, when a letter from Dr. Benjamin Church was intercepted. The coded message was suspected to be vital information for the British. However, the revolutionaries were unable to decipher it. Later on, Elbridge Gerry (who went on to become the fifth Vice President) and Elisha Porter solved the cipher. The message proved Benjamin Church guilty of transferring information to the Tories and he was exiled. The cipher was a simple one where each correspondent had a code book. Each word of the message was replaced by a number representing its place in the book. For example, 3.7.9 meant page 3, line 7 and word 9 in the code book.

Ciphers were also employed by the revolutionaries during the American Revolution. General George Washington was informed about the British troops and their movements around New York City through Samuel Woodhull and Robert Townsend. Their code was a cache of numbers which was substituted for words. This was written by Major Benjamin Tallmadge and also used invisible ink for increased security.

## Rapid advancements

James Lovell is considered as the father of American Cryptology. He was true to America and helped the revolutionaries decipher the British codes. This helped them gain victory in the war, especially during the final stages.



US army cipher device

Interestingly, when former Vice President Aaron Burr and his assistant General James Wilkinson were planning to colonize Spain, they were confused regarding possession of the annexed state. Would it belong to the United States of America or Aaron Burr? Taking advantage of the situation, General Wilkinson who was a Spanish agent changed Burr's coded message, giving a portrayal of Burr's intentions of making Spain his

own country. The letter was intercepted and was brought to the notice of President Thomas Jefferson. Burr was acquitted after a trial but sadly, his reputation was tarnished forever.

Fascinated by cryptography, President Thomas Jefferson then went on to create a cipher of his own. However, it must be noted that a similar cipher was used by the US Navy a few years before the Jefferson cipher was invented.

## Jefferson's Wheel

In 1795, Thomas Jefferson invented the 'wheel cipher', though he didn't use it much. It basically consisted of a set of wheels where the letters of the alphabet were randomly distributed. The key is the order of the wheels on the axle. Thus, the message could be coded by arranging the letters along the axis of the axle so that the messages are formed. Any other row of such letters can also be employed as ciphertext. But to decode, the recipient must align the ciphertext to the rotational axis. Without the knowledge of the arrangement of symbols on the wheels, a plaintext of any length can be churned out. This makes the cipher a secure one for the first use. But using the same wheels in the same order for multiple occasions can be used for statistical relays.

Developing on the wheel cipher, Colonel Decius Wadsworth created a set of two disks, where one was embedded inside the other, way back in 1817. The outer disk had the letters of the alphabet and numbers 2-8

while the inside disk featured only the alphabets. The disks were put together at a ratio of 26:33. To code a message, the inner disk is turned until the required letter is at the top position along with the number of turns required to result in the ciphertext. The ratio ensures that the repetition doesn't happen till all the 33 characters of the plaintext have been used. Sadly, Wadsworth wasn't credited for this design because Charles Wheatstone devised a rather similar machine and took away the limelight.



WWII Enigma Cipher Machine

## Codes and World War II

America's involvement in the Second World War was the result of an intercepted cipher. Known as the 'Zimmermann Telegram', this cable from the German Foreign Office was delivered through Washington to its ambassador Heinrich von Eckardt in Mexico. This was an invitation for Mexico to ally with Germany and in return, they were promised the chance to invade the United States of America. This was thwarted as the cipher was intercepted, and played a massive role in getting America involved in World War II.

## Overcoming Enigma

During World War II, information could win or lose a battle and cryptanalysis was an integral part of it. The Nazis relied on the Enigma machine which was used for cryptography to secure intelligence which was vital. By this time, mechanical and electromechanical ciphers were widely utilized. After the telegraph, the advent of the radio changed the game and further developments were made when the rotor system of cryptography was used. Radio completely changed the landscape and French radio stations intercepted most German radio transmissions, though the latter used a double columnar transposition called *Ubbi*, which was not very strong.

Germany utilized cryptography immensely in all its forms. The Polish Cipher Bureau helped with decoding the detailed structure of the Enigma using mathematics while documentation was given by French military intelligence. This was perhaps the biggest breakthroughs in the history of cryptanalysis. The British Cryptanalysis Department was made up of chess masters and mathematicians, who helped in decoding of the Enigma's encryption. But these officers were not allowed to openly show their achievement at having broken the ciphers fearing the Germany would claim that Britain had not waged a fair war.

## Winning wars

The US Navy was able to break into the codes of the Japanese Navy and this helped them gain an edge in the Battle of Midway, ensuring victory. Later on, the highly advanced Japanese diplomatic cipher system with an electromagnetic stepping switch machine, nicknamed "Purple" by the Americans, was broken. The Americans termed the intelligence emerging from Purple as Magic.

The German military also began attempts in earnest, creating “Fish” ciphers, and finally invented the world’s first ever programmable computer, the Colossus, in aid of cryptanalysis.

Finally, America created the MI-8 in 1917, a unit dedicated to cryptanalysis. This organization analysed all secret messages, links and codes. This was largely successful after WW I but in 1929, it was decided to have it shut down as it was found inappropriate to “read others’ mail”. An American couple, William Fredrick Friedman and his wife, Elizabeth Smith were a famous couple in cryptography, having devised new methods to solve ciphers through frequency counts and superimposition.

## Codes in words

A lot of literature is the perfect foil for cryptography. Take Edgar Allen Poe’s cryptographic uses, for example. He used many systematic methods to decipher in the 1840s. He also went on to advertise his abilities in the Philadelphia paper Alexander’s Weekly (Express) Messenger and invited ciphers to be submitted, of which he solved almost all. Poe wrote a comprehensive essay which was quite handy for amateur British cryptanalysts who were involved in breaking German ciphers during World War I. That apart, ciphers played a vital role in his famous story, “The Gold-Bug”.



Raphael's Sistine Madonna  
(c. 1513-1514)

Arthur Canon Doyle’s Sherlock Holmes also used simple ciphers to solve several mysteries. When a cipher was delivered to Holmes, he used the Almanac as the “codebook” to decipher the message.

The earliest and most famous of literal ciphers were described by Sir Francis Bacon. The bilateral cipher uses two fonts,

one ordinary and the other specially cut. The difference between these two fonts is so minute that it can only be noted through a powerful magnifying glass. Initially, italics were used to communicate but being more ornate; they offered a perfect disguise for the plaintext.

Shakespeare’s four folios use this cipher throughout the text, where almost entire scenes have been added. The bilateral cipher was not just

restricted to Bacon and Shakespeare but was used for over a century after their deaths. However, this cipher is impossible to use now due to standardization of the text for publishing which is pre-set.

## Musical ciphers

Quite uncommon was the musical cipher where musicians exchanged notes of music for the letters of alphabet. Two people can converse by merely playing a few notes on the piano. They can be involved to a point of no return. The trick is to mildly alter the composition without changing the arrangement. However, the scope of musical cipher is largely limited. It was believed that many such music compositions by Sir Francis Bacon are existent even now.

## Picture this

The pictorial cipher was quite popular and is basically a picture or diagram which reveals more than its obvious meaning. Egyptian symbolism is rich in pictorial ciphers and diagrams of alchemists and philosophers reveal hidden layers. The detailing in the images can make a cipher of this sort rather complicated. For example, codes can be hidden in the ripples on the surface of water or by the number of stones in a wall. Montaigne's Essays use the pictorial cipher rather wisely. The initial B is formed with the help of two arches and an F aided by a broken arch. Sometimes, these pictures are accompanied with a key. This was used extensively in art to show meaning in the depths of colour.

Popular examples of pictorial ciphers include Leonardo Da Vinci's Last Supper which created a furore over the relationship between Jesus and Mary Magdalene as well as the hotly debated fifth finger on the Pope's hand in Raphael's Sistine Madonna. The same artist chose to add a sixth toe on Joseph's foot in his painting, Marriage of the Virgin. These are brilliantly concealed cryptograms.

## Religious ciphers

Acroamatic ciphers were the religious and physical writings of all nations, where allegories and parables are often the sources of cryptography. Since time immemorial, parables and allegories have been used to present the truth in an easy-to-understand manner. This is a pictorial cipher drawn in words and understood through symbolism. It includes the Old and New Testaments of the Jews, the works of Plato and Aristotle, Homer's Odyssey

and Iliad, Aesop's Fables and Virgil's Aeneid. This is the most subtle cipher and can be interpreted in many different ways.

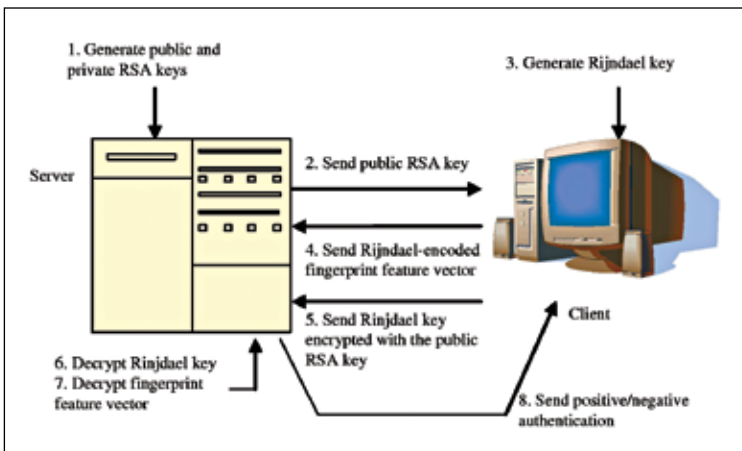
## Indian elements

Cryptography was mentioned in the Kamasutra, written way back in the 4th century AD. Based on the manuscripts, it recommends that women should study 64 arts and one of them was the art of secret writing (mlecchita-vikalpa), so that they can hide the details of their liaisons. Early prototypes included a simple system where random pairing of the alphabets was followed by substitution with letters from a cipher.

The government also employed secret codes to communicate with spies distributed across the country. Apart from ciphers based on substitution, they also relied heavily on spoken or sign language communication.

## Modern cryptography

While Vigenere's cipher was unbeatable for many, many years, all that was about to change as Charles Babbage added his unique element to cryptanalysis. An independent and wealthy Englishman, Babbage was credited for his work in computer science. He hacked systems in search of a repeated sequence of letters. Babbage's indigenous technique created waves and took cryptography to the next level, introducing mathematics in place of words.




Public key cryptography

After World War II, most cryptography was mathematical in approach, thanks to the easy availability of computers and the discovery of the internet as a communication tool. Shannon was the frontrunner of modern cryptography, relying mostly on mathematics. His extensive work during WWII and publication in the technical journals laid a theoretical base for mathematical cryptography. This was immediately utilized by secret Government organizations like GCHQ and NSA.

In 1970s, the Data Encryption System (DES) made its entry, proposed by a research team at IBM, in a bid to secure electronic communication. This would immensely help business facilities for banks and other financial concerns. After creating a furore, the DES soon became obsolete and was replaced by the Advanced Encryption Standard (AES). Rijndael, a program developed by two Belgian cryptographers were believed to be secure and are still used today. However, this can be highly vulnerable to brute attacks.

## Back to the future

The biggest recent development occurred when the Public Key was developed, where a new method of key distribution was introduced. The Diffie-Hellman key exchange created a new class of coding algorithms, using asymmetric keys. Before that, all the keys were rather symmetric in approach, which brought with it a host of problems including the availability of secure channels, especially as the number of participants increase. But the asymmetric key utilizes mathematical keys to decrypt the encryption. By designing a private and public set of keys, the requirement of a secure channel is diminished.

These early ciphers laid the foundation for their modern day counterparts, much more advanced and better suitable for delicate transactions. Right from credit card transactions to internet safety, cryptography is now an integral part of our lives. And its importance has only been multiplied in recent times. 



# SIMPLE CIPHERS EXPLAINED

Starting off with the basics, we take you through the elementary types of ciphers out there

If u havcme dis far rdng dis buk, vblve u wudfind dis txt prtysz 2 read. So was that really that tough to understand? We hope not. After all, we do it everyday – saving characters in our text messages on mobile phones and during chat to type faster. But it's not a new concept. The current generation can't take credit for this invention; it's not our invention.

Our forefathers used it in telegrams. Back when sending telegrams was charged per character, they'd try to squeeze up characters to form another word which could easily be related to the word that it was meant to replace – much the same way we abbreviate text messages today.

But does that hide the message? Not at all. After all, you did understand the first sentence of this chapter, didn't you? So did the cipher fail to do its job? Well, no, it didn't; it wasn't a cipher at all. It was a code.

## Codes

History is full of stories of encryption. It doesn't matter if you're a fan of Sir Arthur Conan Doyle or Dan Brown or think you were a World War II hero in your past life, the fact that you're reading this generates a high probability that you already know about ciphers to some extent.

If you refer to day-to-day references, you can easily alter your focus and blur the line between codes and ciphers, seeing them as one and that wouldn't be all that wrong. Changing the names of your friends to "Vampire", "Sun" and "Scooter" is a common practice among those who want to hide the numbers of their friends from other unwanted users by confusing them. If you've used such a technique, chances are that you would recommend it to close friends as well.

But were you actually successful in keeping the identities hidden? Maybe, maybe not! Let's assume that you actually have those names in your phone's address book. What are the chances someone else could understand what "Vampire", "Sun" and "Scooter" stood for? If you're a fan of the (in)famous "Twilight" series of books, Vampire could be a very special friend. If you have a friend who eats a lot of non-vegetarian food you might use the same name for him. If your friend's real name is Sunny, 'Sun' would obviously be in reference to him. Scooter too, in the same manner can be related to someone with that nickname which stuck with them due to an incident, or maybe it was an accidental pet name you discovered in childhood.

The question that remains unanswered is "What's the probability that someone else could understand it?" If you paid attention, you'd notice that all the names were changed based on the real life experiences. Obviously then, close friends/family would be curious to know which name in your phonebook referred to which friend, and if they were good enough sleuths, they'd figure it out.

Now, let's get into the technicality of the subject and bring up a few terms. While we won't talk about "plaintext" and "ciphertext" again, it's

noteworthy that when working with codes, the terms are “decoded” (or “plaintext”, again) and “encoded” in that order.

Another term when it comes to coding in the context of cryptography is “codebooks”. In the simple case that we just presented, your life history serves as a codebook. To help you better understand, here’s another simple example:

Ethan and Fabio want to communicate with each other. But there’s a problem – Ethan knows only English while Fabio knows only French and can’t speak or understand English. Ethan too can’t understand French. In this case, Ethan would require a French->English dictionary so that when Fabio speaks in French, he could look up the words in the dictionary and find their meanings in English. Similarly, Fabio will need an English->French dictionary so that when Ethan speaks in English, he can lookup the words in the dictionary and understand their meanings in French. In this case, both the dictionaries serve as codebooks, using which one can find the “decoded” form of an “encoded” word. Technically, a codebook is a simple mapping of encoded-to-decoded words. If you were to transform all names in your phonebook in a way that no one could easily find out what name stood for which person, you might need a codebook other than your life history.

That was one form of encoding. The other form of encoding is called a “one-time” code. This form of code, as the name suggests, is used once only. One of the best examples of such code would be the military’s use of code-names for teams of soldiers. Let’s again consider an example: Assume that there’s a military general with three teams of soldiers. On the first day, he assigns names as: Team 1 => Alpha, Team 2 => Bravo and Team 3 => Charlie. Then he sends them out for patrol. The teams communicate using wireless radio devices whose signals can be intercepted. So when Team 1 has to talk to Team 2, a person from Team 1 says “Alpha to Bravo” to signal that the message is meant for Team 2 from Team 1. Though Team 3 also gets the message, they understand that the message is not for them. The next day, names are changed and Team 1 => Zebra, Team 2 => Delta and Team 3 => Tango. If Team 1 once again wants to convey a message to Team 2, they would say “Zebra to Delta”. If these messages were also being intercepted by enemy soldiers, they wouldn’t be able to determine the total number of teams (notice the first naming was done in alphabetical order, while the second was random). Such usage of codes is an example of a one-time code.

One-time codes are utilized only once (depending on what is defined as “once”) and are often used for messages which are to be “broadcasted” and

then “delivered” directly to the destination. These codes are discarded after their use and a new code is formed. Also, all parties who are to use the code must know the code and its meaning before the scheduled (or expected) broadcast of the message.

Till now we’ve spoken about condensed text messages, examples of codebooks and one-time codes. You still don’t know what a cipher is though, and where do you draw the line between ciphers and codes?

## Codes and Ciphers – the difference

Ciphers are a way to scramble messages so that other people don’t understand them. They differ from codes in multiple ways. Let’s list them:

1. Codes work on larger chunks, usually words. Ciphers work on lower levels. Classical ciphers work with individual letters while modern ciphers work with individual bits.
2. while ciphers are algorithms. Codes involve mapping from one language to another (encoded to decoded) and vice versa. On the other hand, ciphers are mathematical algorithms, often very complex. As such, codes would need a codebook while ciphers won’t.
3. Cryptanalysis is usually not possible for wisely encoded messages. Since there’s no mathematical algorithm involved, it might not be possible to find patterns in the code. For example, a code might specify that “all words starting with a vowel are to be discarded first before beginning the actual decoding process.” Such a mechanism makes it nearly impossible to guess the second step involved while still containing a fully valid message (using a codebook where all encoded words begin with a consonant). Cryptanalysis depends on analyzing scrambled messages, but that’s not possible with codes since they may not be broken using techniques of cryptanalysis.
4. Codes depend on the person who compiles the code and the codebook itself. The strength of the code depends on the technique he utilizes as well as his intelligence. If the code designer wasn’t intelligent enough, the code’s strength would be weak. On the other hand, ciphers depend on already designed mathematical procedures and their design doesn’t rely, time and again, on humans.

Above all those differences is the fact that all ciphers require at least one key, while codes don’t. While some may say that the encoding-decoding process also requires a codebook which is analogous, they’re not similar enough to be compared. The process of coding is based on mapping– you

have an encoded word and you have a decoded word and they're in a one-to-one relationship based on a codebook. The process of ciphering is based on calculation- you have mathematical algorithms which can scramble and unscramble messages which would require the correct key.

Modern day ciphers which work on bits didn't just come out of the blue. They were based on previous research, which in turn was based on even older research.

## Simple Ciphers

### Substitution Cipher

This is one of the simplest ciphers possible. The process is simple and apparent from the name - you substitute things. "But what things?" is a more important question. The most common 'thing' to be substituted is a single letter. One of the most common methods substitution ciphers use is called ROT13. Here you rotate the alphabets by 13. So 'A' becomes 'N', 'B' becomes 'O' and so on until 'M' becomes 'Z'. Since the English alphabet has only 26 letters, ROT13 makes the mapping easily reversible - i.e. 'N' becomes 'A', 'O' becomes 'B' until 'Z' becomes 'M'.

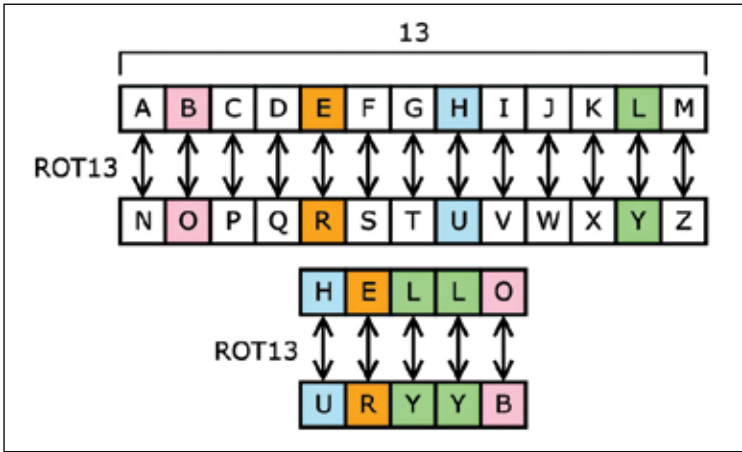
Using the cipher, the word 'HELLO' becomes 'URYYB' and 'DIGIT' becomes 'QVTVG'. Notice that 'T' was substituted with 'G' because ROT13 utilizes the same algorithm for encrypting as well as decrypting the message. Thus, it provides almost no security against analysis.

One can however utilize other rotation techniques too for substitution for single letters which may include rotation techniques other than ROT13 - for example by rotating only 3 times instead of 13. That would map 'A' to 'D', 'B' to 'E' until 'W' becomes 'Z' and again 'X' becomes 'A'. Another method that can be used is random mapping of letters, something like:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Q	F	Y	M	Z	X	L	U	A	E	I	J	V	D	C	R	N	T	H	W	P	S	G	B	K	O

In such a scene, the word 'HELLO' becomes 'UZJJC' and 'DIGIT' becomes 'MALAW'. While this is better than ROT13 where same function could be used for encrypting as well as decrypting the message, it still doesn't provide enough security against cryptanalysis because simple patterns of usage of words can easily signal the ciphering.

If in case it slipped your eye, the 'random' substitution cipher we just devised above is not much of a cipher but closer to a code. The reason resides in the fact that the mapping is completely random and for ciphering or



ROT13 mapping

deciphering (well, we should already call it encoding and decoding), the table must be present. A person receiving a message ciphered using the map presented in the table would not be able to decipher it without the complete table available with them. Why do we present it if it's just a code? Well, we do so to show the fine line which separates coding from encryption. We do so because we think it's fascinating to witness how organizing a random shuffle using a simple number turns a code into a ciphering technique; how it starts to involve calculations instead of maps!

What we demonstrated is a simple substitution cipher which involves rotating the alphabets by a number. However this is just one example of how substitution ciphers can work. There are other ways in which substitutions can be done. One of the more complicated ciphers is Vigenère cipher which utilizes a table of substitutions.

This cipher is known as a type of 'polyalphabetic' cipher as a single letter in plaintext can take multiple values in ciphertext. It utilizes the rotation technique we just described. As you can see, the table contains 26 rows each for each character in the alphabet where the position of the alphabet determines the number of rotations done for the row. The way to use this cipher follows:

1. First get the plaintext. Let it be "ILOVEDIGITMAGAZINE".
2. Decide a keyword which will be used to encrypt the plaintext. Let's use 'TRACK' as our keyword.

3. Now we begin the encryption process:

- Take the first letter of the plaintext: 'T'.
- Take the first letter of the keyword: 'T'.
- Find the corresponding of letter for 'T' in the row beginning with 'T'.  
It is 'B'.

Now you have to repeat the process again from step 3.a to 3.c. This time you would take the second letter of the plaintext and find its corresponding alphabet in the row marked by the second letter in the keyword. Hence, the letter to be encrypted is 'L' and the row you have to see its replacement in is 'R'. The result is 'C'.

Now, the keyword is 5 characters long but the plaintext is longer than the keyword. So how do you encrypt the 6th character? Well, you use the first character of the keyword again. To encrypt the 7th character of the

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Vigenere square diagram

plaintext, you use the 2nd character of the keyword and so on. Hence, the letters of the keyword are used in a cyclic manner. Hence if you were to encrypt the complete plaintext using the given keyword, it would become: BCOXOWZGKDFRGCJBEE. So? Is the ciphertext now traceable back to the original plaintext? Well, no. Not at least without the keyword. This brings us to two interesting conclusions:

The strength of the keyword would determine how the encrypted text turns out. For example, if we chose a small keyword such as 'HIM' or 'ME', it would be much more easier to track it back to the original plaintext. Not only the length of the keyword but also the randomness of the characters in the keyword improve the strength of the cipher. For example, if we used a keyword such as 'DDDDDD', would it increase the strength of the cipher? No, it would actually weaken the cipher as it would just boil down to 'D' being used as a keyword which would directly translate to a rotation of 3 on the alphabet – pretty weak, isn't it? On the other hand, if we use "CRAZYFOX" as the keyword, it strengthens the cipher because it's longer and has no repeating characters. To use "XARCZYFO" would be even better because this makes the keyword itself gibberish and more difficult to predict. These principles (longer keyword, random character distribution in keyword and non-dictionary keywords) still happen to be important factors governing the strength of the encryption.

Spaces in the plaintext could make the ciphertext weaker. If you notice, our plaintext didn't have any white-spaces between them. If they had, the ciphertext would contain spaces too, or some other character to represent white spaces which make the text more guessable than when it does not.

## Transposition Cipher

The word says it all – it uses transposition techniques to encrypt the plaintext. The word 'transposition' means to change the position. This cipher, however doesn't work on character level, but on the word level. In this case we do not change the letters themselves but the way we read them. Let's take our previous example with "I LOVE DIGIT MAGAZINE" as our plaintext (alright, we added spaces). When we write the same sentence backwards, it becomes "ENIZAGAM TIGID EVOL I". This is certainly more difficult to figure out than the plaintext but breaking it's not a big deal by any means. If we were to eliminate the spaces, it would be "ENIZAGAMTIGIDEVOLI", making it a bit more difficult. There are other more popular transposition ciphers which include:

**Rail fence cipher:** We basically write the plaintext in a way which creates linear patterns in a spiral way. For example: if you have a look at your (US english) keyboard, you would see that the character sequence “QAZSEDCFTGBHUJMKOL” are in a spiral sequence. But when you read them in order the letters appear in rows from top to bottom, it becomes “QETUOASDFGHJKLZCBM”. Now we use the same technique to encrypt “ILOVEDIGITMAGAZINE”. It would look like:

I                      E                      I                      G                      N  
     L          V          D          G          T          A          A          I          E  
         O                  I                  M                  Z

Reading it on rows, it becomes “IEIGNLVDGTAAIEOIMZ”. In this case, the key is ‘3’ as we are using 3 rows to do the transformation. Try to do it with key of ‘4’ and you would find that the ciphertext changes.

**Route Cipher:** In this case, the plaintext is laid out in a grid and then written as a sequence of characters which is derived from a visible sequence over the grid. Let’s assume the same plaintext and lay it in vertical columns on a 6 x 3 grid:

I	V	I	T	G	I
L	E	G	M	A	N
O	D	I	A	Z	E

Now read it “inward spiral beginning top right corner”. The ciphertext becomes “IGTIVILODIAZENAMGE”. However, if you’re using a grid of 5 x 4, the grid would become like:

I	E	I	G	N
L	D	T	A	E
O	I	M	Z	
V	G	A	I	

The last two cells on the lower rows remain empty. Those can be filled with any nonsense which the interpreter on the other end can easily understand and discard. In this case the width of the grid can serve as the key for the encryption. The number of rows can be calculated using simple mathematics given the padding was done.

**Columnar Transposition:** This technique is a bit similar to the router cipher. Here too we lay out the plaintext in columns but the way we would

convert that grid into ciphertext differs from the route cipher technique. The process is as follows:

1. We first select a keyword. Let the selected keyword be “DUFFER”.
2. Now, we remove the repeating letters from the keyword. Now, the keyword becomes “DUFER”.
3. Next we re-arrange the letters of the keyword we obtained in the previous step. It becomes: “DEFUR”.
4. We give each letter a number in sequence. D -> 1, E -> 2, F -> 3, R -> 4, U -> 5.
5. Now we take the keyword from step 2 and replace the letters with numbers we got in step 4 (last step). So the keyword now becomes “1 5 3 2 4”.
6. We now lay out a table with 5 columns. The number sequence we obtained in the last step are placed in the top row serving as heads.
7. We then lay out our message in the table linearly. The table would look like:

1	5	3	2	4
I	L	O	V	E
D	I	G	I	T
M	A	G	A	Z
I	N	E	X	X

Please do note that ‘X’ in the last row serves as padding to avoid irregularities in the encryption. When decrypting the ciphertext on the other end, it would be easily detectable and thus can be discarded.

8. Next we write the text in the column number 1 (from top to bottom) followed by text in column number 2 (again from top to bottom). The ciphertext comes out as: “IDMIVIXOGGEETZXLIAN”.

The keyword “DUFFER” can still be used to decrypt the message. Try doing that yourself if you want to put your grey matter through its paces.

With that we round up the simple ciphers and will now try to figure out more modern techniques which provide higher protection than these classical and simple (and also weak) ciphers provide. Nonetheless they are important because they help us understand how we have progressed so far to a place and time where clever cryptanalysis techniques combined with superfast computers fail to get back the plaintext within any practically meaningful time limits.

## Modern Ciphers

We’ve spoken about some of the simple ciphers till now which worked by

substituting or shuffling the letters from the alphabet. It's time we now move on to the digital age where machines do the difficult calculations required to encrypt and decrypt messages, while the work of us mortals has been reduced to figuring out the techniques which would produce stronger ciphers. One of the biggest differences between traditional or classical ciphers, and modern ciphers, lies in the smallest unit which they can operate on. Classical methods had a single letter as the smallest unit, while the modern methods have a single bit as the smallest operable unit.

As machines evolved and the need to communicate between machines grew, standardisation of protocols could not have been avoided for too long. ASCII emerged as the most popular text encoding technique. It worked by treating every number as an integer (and that is a type of simple coding technique again, but was meant to be open) and the integers in turn could be represented as a sequence of bits – hence the modern techniques (which are mostly developed for computers) can work with bits.

Modern techniques of ciphering and deciphering heavily depend on numbers (all text is actually numbers in ASCII, remember?) and can be classified into two broad categories: symmetric and asymmetric.

**Symmetric algorithms:** A method in which the same key is used for encrypting as well as decrypting the text is called symmetric algorithm. All the classical ciphers we've illustrated thus far would fall under this category because you would always use the same key to encrypt as well as decrypt the message. Some of the better known examples of symmetric algorithms are DES, 3DES (or Triple DES), AES and Blowfish.

**Asymmetric algorithms:** A method in which the key used for encrypting the message cannot be used to decrypt the message. The key which is to be used for decrypting the message has to be different from the key used to encrypt the message and the relationship between the two keys is complex enough that the second key cannot be derived from the first key. The most popular example of this breed is 'RSA'. If you ever check the SSL certificates of websites you visit, chances are you've read it, unless of course you already know about RSA.

Asymmetric algorithms gain importance over symmetric algorithms because of the fact that symmetric algorithms depend on the same key for encryption as well as for decryption. Since the key is also to be transmitted to the receiver, any person who is able to intercept the encrypted message as well as the key would be able to decrypt the message, thus rendering encryption completely useless. Since with Asymmetric algo-

rithms, the key used for encryption cannot be used for decryption, it's safe to transmit the public key (the one which is to be used to encrypt the message). Any person who wants to send back a secure message can encrypt the message using your public key and you would be able to decrypt it using the private key!

## DES

There is no way we can explain DES in detail within the limits of this book. Hence we would provide you with an introductory explanation of this algorithm. DES is actually an acronym for Data Encryption Standard.

DES is a block cipher which works by running an operation for 16 rounds on blocks of 64 bits. On each round, it takes the second half of the plaintext (32 bits) and transforms it into different text using a pre-designed algorithm. It then does an XOR operation of the result with the first half of the plaintext. The result of the XOR operation is processed by the algorithm (denoted as 'F' in the diagram) in the next step, while the second half is XORed with the result of the 'F' function in the next step. The method to get the plaintext is to reverse the procedure – to go from 'FP' towards 'IP' and reverse the algorithm.

## RSA

The algorithm was designed by Ron Rivest, Adi Shamir and Leonard Adleman and it's their last names which make up the name of the algorithm. The RSA algorithm is given in following steps:

1. Two prime numbers  $p$  and  $q$  are taken.
2. A new number  $n$  is calculated by multiplying  $p$  and  $q$   

$$n = pq$$
3. Another number  $\phi(n)$  is generated by using the formula:  

$$\phi(n) = (p - 1)(q - 1)$$
4. Another integer  $e$  is to be found such that  $e$  lies between 1 and  $\phi(n)$ . Also,  $e$  and  $\phi(n)$  must be coprimes. Mathematically  

$$1 < e < \phi(n) \text{ where } \gcd(e, \phi(n)) = 1$$

(gcd = Greatest Common Divisor)
5. Another number  $d$  is to be found such that  

$$de \bmod \phi(n) = 1$$

i.e. When the product of  $d$  and  $e$  is divided by  $\phi(n)$ , the remainder should be 1.

The numbers 'e' and 'n' make up the public key, while 'd' and 'n' make up the private key. The calculation is given as follows ('m' is the message to be encrypted and 'c' is the ciphertext produced after encryption):

For encryption:

$$c = m^e \pmod{n}$$

For decryption:

$$m = c^d \pmod{n}$$

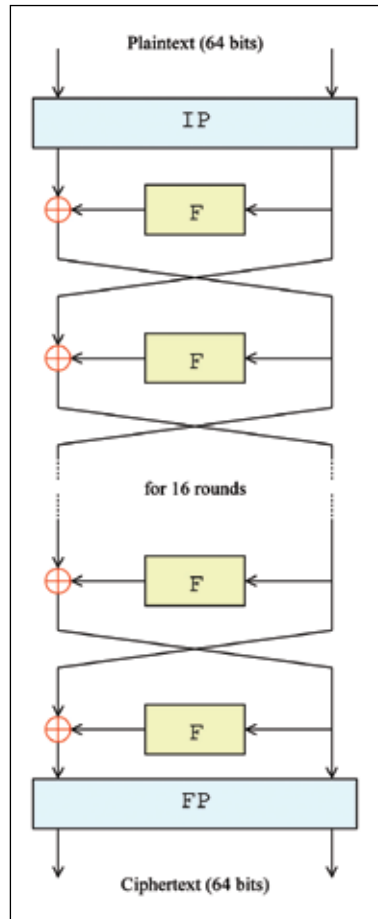
Note: the character ^ refers to the power operator as in  $2^2 = 4$  and  $2^3 = 8$

Now, this is applicable for numbers and messages are not always numbers. However, with digital messages it's different because any and everything is represented in binary form which can be directly translated to a number.


The strength of the cipher depends on how difficult it's to guess the numbers p and q. Once p and q are known, it would not take too much effort to find the rest of the numbers. Since the product of p and q is n and n is a part of public key which is used for encryption and can be well known by third

parties, the only way to know p and q is to find out the factors of n. Interestingly enough, n can have only 4 factors: 1, n, p and q. The first two are applicable for any number and thus one has to find p and q. The strength of RSA, as such, depends on p and q. Hence they both should be very large prime numbers. This makes the factorization of n extremely difficult (since there are no easy ways known to factorize the products of large prime numbers).

Algorithms are what protect the secrecy of the message and the strength of the cipher depends on two things: the key used and the algorithm used.



Overall DES Algorithm

If either of them is weak, the message can be intercepted by third parties. While specially designed algorithms were used as way back as ancient Rome, they wouldn't work today as they would be too simple. In the modern world, it all boils down to the strength of the keys used, and millions if not billions are spent on military grade encryption, for financial institutions, software and more. 



# USES OF CRYPTOGRAPHY

Let's now discuss the expanded role and usefulness of Cryptography in modern times

**T**he crux of what you've learned so far is that cryptography is the art of writing or storing information in such a way that it's revealed only to those who need to see it and hides it from all others.

Long before the information age, cryptography was used only to ensure secrecy of information. Encryption was used to ensure confidentiality in communications by spies, military leaders and diplomats. The Egyptian hieroglyphs, the scytale transposition cipher used by the Spartans of Greece, waxed seals and different physical devices to assist

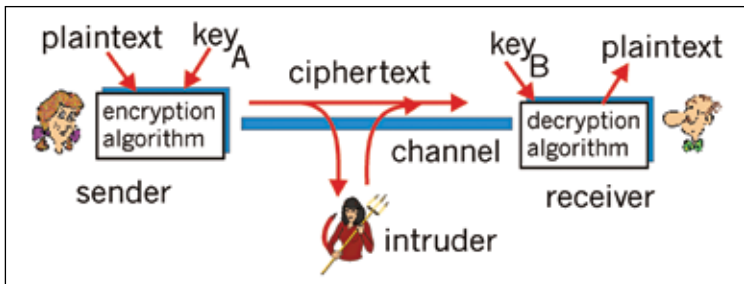
with ciphers were used throughout history right up to modern times. These devices underwent further changes when computers and electronics came into the picture, immensely helping in cryptanalysis. Cryptography has become more mathematical now and also finds applications in day-to-day security. It helps you safely transfer or withdraw money electronically and you'd be hard-pressed to come across an individual without a credit or debit card. The public-key encryption system introduced the concept of digital signatures and electronic credentials. Cryptography has a definitive existence in our lives today and the whole system will crumble in its absence.

Let's now discuss the varied uses of cryptography in modern times and its intersection with computer science.

## Secrecy in transmission

The major goal of cryptography is to prevent data from being read by any third party. Most transmission systems use a private-key cryptosystem. This system uses a secret key to encrypt and decrypt data which is shared between the sender and receiver. The private keys are distributed and destroyed periodically. One must secure the key from unauthorized access, because any party that has the key can decrypt the encrypted information. Alternately a key-generating-key, called a master key, can be used to electronically generate a one-time session-key for every transaction. The secrecy of the master-key should be maintained by all parties privy to the information. The disadvantage of this method is there's too much hope riding on the master-key, which if cracked, collapses the entire system.

A better method is to use a public-key cryptosystem. In this system, data can be encrypted by anyone with the public-key, but it can be decrypted only by using the private-key, and data that is signed with the private key



How an encrypted transmission can be intercepted

can be verified only with the public key. With the development of public-key systems, secrecy can be maintained without having to keep track of a large number of keys or sharing a common master-key. If, say, Alex wants to communicate with Neil, she first generates her public/private key pair and sends the public key to Neil over a non-secure channel. Neil uses that key to encrypt information and sends it back to Alex. Only Alex has the private key with which she can decrypt the information. Anyone who intercepts the public key or the encrypted data can't decrypt the message due to the protocols followed during information transfer.

## Secrecy in storage

Storage encryption refers to the application of cryptographic techniques on data, both during transit and while on storage media. Storage encryption is gaining popularity among enterprises that use storage area networks (SANs). Secrecy in storage is maintained by storing data in encrypted form. The user has to provide the key to the computer only at the beginning of a session to access the data and it then takes care of encryption and decryption throughout the course of normal use. Hardware devices can also be used for PCs to automatically encrypt all information stored on disk. When the computer is turned on, the user must supply a key to the encryption hardware. The information is plain gibberish without its key thus preventing misuse if the disk is stolen.

Multiple ciphers can be used for individual files and folders. The ciphers and keys should be changed frequently to ensure security of data. However, if the user forgets a key, all of the information encrypted with it makes no sense and is rendered useless. This is why backups of encrypted information are advised to be stored in plaintext. The data is only encrypted while in storage, not when in use. This leaves a loophole for the attackers. The system is vulnerable to a security breach if the encryption and decryption are done in software, or if the key is stored somewhere in the system.

## Integrity in transmission

We can use cryptography to provide a means to ensure that data is not altered during transmission, i.e. its integrity is preserved. In electronic funds transfer, it is very important that integrity be maintained. A bank can lose millions if a transaction is illicitly intercepted. Cryptographic techniques are employed to prevent accidental or intentional modification of data during transmission, leading to erroneous actions. One of the ways

to ensure integrity is to perform a checksum on the information being transmitted and to transmit the checksum in an encrypted form as well. The information is received on the other end and again checksummed. The transmitted checksum is decrypted and compared with the previous checksum. If the checksums agree, the information is most likely unaltered. The problem with this scheme is that the checksum of the original message can be known and another message with the same checksum can be generated and sent instead of the original one. This problem can be overcome by using a public-key cryptosystem. After generating the public-key/private-key pair, if we throw away the private-key and use only the public-key to encrypt the checksum, the checksum becomes impossible to decrypt. In order to verify the checksum, we generate a new checksum for the received information, encrypt it using the public-key and match it with the encrypted checksum. This is also known as a one-way function as it is hard to invert.

## **Integrity in storage**

Integrity in storage had been ensured by access control systems with lock and keys and other guards to prevent unauthorized access to stored data. The existence of computer viruses has changed the scenario and the need of integrity against intentional attack has become a problem of epic proportions. Cryptographic checksums to ascertain validity of stored data are of help here. As in the case of transmission, a cryptographic checksum is produced and compared to the expected value. However, storage media are more vulnerable to attacks than transmission channels due to longer exposure and larger volumes of information.

## **Authentication of identity**

Authentication is the process of verifying if the user has enough authority for data access. Simple passwords are used to identify someone. You must also have seen in classic gangster movies, the exchange of keywords to prove identity. Cryptography is similar to the practice of providing passwords for identity authentication. Modern systems use cryptographic transforms in conjunction with other characteristics of individuals to provide more reliable and efficient authentication of identity. Many systems allow passwords to be stored in an encrypted form, with read access available to all programs which may use them. Since passwords are not stored as plaintext, an accidental leak of data doesn't compromise the system's security.

Passwords are analogous to the key in a cryptosystem that allows encryption and decryption of anything the password has access to. The principal element of this system is the password selection process. And that's a whole other subject that we can't cover here. But in a nutshell, the longer the password, the more random it will be and the harder it is to guess. So if you think it's easy for you to remember, you should know that it will be all the easier to crack.

## Credentialing systems

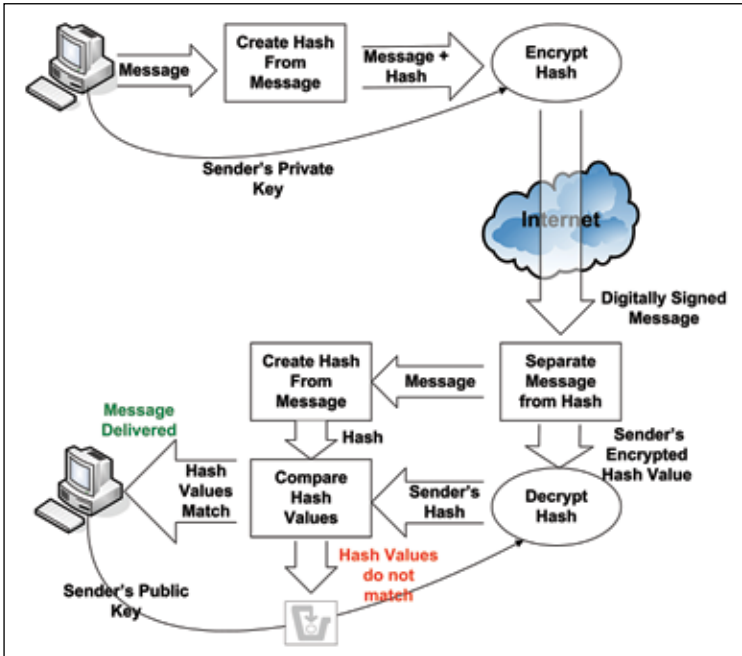
A credential is a proof of qualification or competence that is attached to a person to indicate suitability for something. Suppose you go to a bank for a loan, they check your credentials before approving the loan. Your credentials are checked not only from the paperwork, but also from your past record and your references. Your driver's license and passport are forms of credentials. Progress in the field of implementing electronic credentials has been rather slow. Electronic credentials allow electronic verification of the credence of a claim. It's not a standalone system, but is being used in conjunction with other devices such as smart cards which perform cryptographic functions and store secret information.

CIBIL (Credit Information Bureau (India) Limited) is India's first credit information bureau. It shares credit information with banks, financial institutions and credit card companies and generates Credit Information Reports.

## Digital signatures

A digital signature is a mechanism by which a message is authenticated i.e. proving that a message is coming from a given sender, much like a signature on a paper document. To be as effective as a signature on paper, digital signatures must be hard to forge and accepted in a court of law as binding upon all parties to the transaction. The need for digital signatures arises when the parties dealing in a transaction are not physically close, and the volume of paperwork is high, in other words big business dealings.

Digital signatures can be created using a public key cryptosystem and hashing process. Hashing produces a message digest that is a small and unique representation of the original message. Hashing is a one-way algorithm, i.e. the message can't be derived from the digest. Let's say that Alex is sending a message to Neil. Alex first hashes the message to produce a digest, and then encrypts the digest with her private-key to create her personal signature; the public-key and hash algorithm are appended to it.



Hashing in action

The whole message including the digest is then encrypted using a one-time symmetric-key which is known only to Alex and Neil. Neil decrypts the message using the symmetric-key. He then decrypts the message digest using the public-key. He would then hash the original message using the same hash algorithm (whose name was appended in the message) with which it was previously hashed. If the evaluated digest and decrypted digest match, then the signature has been verified and the recipient would be sure that the message integrity has been preserved.

Another aspect of this system is the non-repudiation of digital signatures. Since the private-key is only privy to the sender, he can't deny signing the message. Also, a digital signature can be verified by anyone using the sender's public-key which is usually included in the digital signature format.

## Electronic money

Electronic information has replaced cash for financial transactions between individuals for quite a long time now. Such a system uses cryptography to

keep the assets of individuals in electronic form. Electronic funds transfer (EFT), digital gold currency, virtual currency and direct deposit are all examples of electronic money. Electronic funds transfer (EFT) is the electronic exchange of money between two accounts through computer-based systems. This includes online payments, debit card payments, ATM withdrawals, direct deposits, wire transfers and the like. Another application of electronic money is in e-commerce, and businesses such as PayPal mediate the transfer. Clearly any attack on such a system would allow wipe out national economies in the blink of an eye. The significance of integrity in such a system is staggering.

The key property of cash is anonymity: when you take money out of the bank, the bank gives you the cash without knowing what you do with that money. The merchant doesn't know who you are or ask for your credentials when you pay in cash. On the other hand, when you buy something with a credit card, you have to tell the merchant who you are, and you have to tell the credit card company who you're purchasing from. Anonymity is not maintained thus failing to protect your privacy. Concerns that anonymity in e-money could encourage tax evasion and money laundering led to demands by various institutions for digital cash to be traceable. This called for an elaborate method of encryption so that the information wouldn't get into the wrong hands.



The man behind it all is Dr. David Chaum. He formulated the blinded signature, a special form of a cryptographic signature that allowed a virtual coin to be signed without the signer seeing the actual coin, and permitted a form of digital money that offered anonymity and untraceability. This form of currency is known as Digital Cash.

## Threshold cryptosystem

Threshold systems are designed to allow use only if a minimum number of parties, exceeding a threshold, agree to the said use. Technically, it means that in order to decrypt a ciphertext a minimum number of parties are required to collaborate in the process. Any less than that won't have sufficient information. For example, if in a bank at least 5 out of 10 people authorize the transaction, only then will it occur. Such systems obviate a single individual acting alone, while at the same time allowing many of the parties to be absent without the transaction being halted.

Most threshold cryptosystems have keys which are distributed into parts. The most common technique for partitioning a key is to form the key as the solution to equations in  $N$  variables. Only if all the  $N$  equations are known, the key can be determined by solving them. If any less than  $N$  equations are known, the key can't be determined since there's at least one independent variable in each equation. The minimum required threshold number can be chosen for  $N$  and the equations can be held by separate individuals. The same general concept can be used to form arbitrary combinations of key requirements by forming ORs and ANDs of encryptions using different sets of keys for different combinations of key holders. The major difficulties with such a system lie in the key distribution problem and the large number of keys necessary to achieve arbitrary key holder combinations.

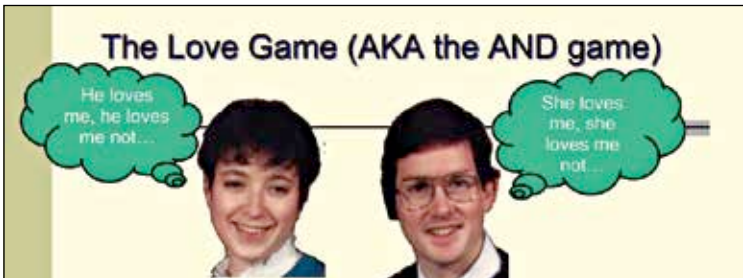
Such systems are mostly employed in organizations with very valuable secrets, such as militaries and the governments. One of the applications is to store the secret information in multiple locations to prevent access to the ciphertext itself and thus prevent cryptanalysis on it.

## Secure multi-party computation

Secure multi-party computation involves a set of parties with private inputs who wish to jointly compute a function of their inputs so that certain security properties (such as privacy and correctness) are preserved. It provides solutions to various real-life problems such as private auctions, distributed

voting, sharing of signature or decryption functions, and situations that require private information retrieval. One popular application of secure MPC (multi-party computation) is solving the Yao's millionaire problem, i.e. two millionaires want to know which one of them is richer but without revealing their net worth to the other.

The millionaires' problem is a secure two-party computation problem. It has been generalized for multi-party computations. In a secure MPC, if no party can learn more from the public function and its result than what can it learn from its own input. For a better understanding of the concept, visit: <http://dgit.in/Qe37wP>




This concept is of great value in the field of cryptography. It has been proved that the multi-party computation problem can be solved if there exist unconditionally secure authenticated channels between pairs of participants. Consider four individuals Alice, Bob, Carol and Dave who want to calculate their average salary without revealing their salary to others. One way to calculate the salary is as follows:

- 1 Alice adds a secret random number to her salary, encrypts the result with Bob's public key and sends it to Bob.
- 2 Bob decrypts Alice's result with his private key. He adds his salary to Alice's message, encrypts the result with Carol's public key and sends it to Carol.
- 3 Carol decrypts Bob's result with her private key. She adds her salary to Bob's message, encrypts the result with Dave's public key and sends it to Dave.
- 4 Dave decrypts Carol's result with his private key. He adds his salary to Carol's message, encrypts the result with Alice's public key and sends it to Alice.

- 5 Alice decrypts Dave's result with her private key. She subtracts the random number from Step 1 to recover the sum of everyone's salaries.
- 6 Alice divides the result by the number of people (four, in this case) and announces the result.

This way no one knew anybody else's salary and the function to calculate average salary was successfully computed. More such examples can be seen at <http://dgit.in/VO6ZU8>.

The aforementioned applications of cryptography help us understand that its use transcends almost all aspects of human dealings. Cryptography ensures security and integrity of information and prevents misuse of data by unauthorized persons. It also makes our lives convenient by providing such instruments as electronic cash and digital signatures. It was used by early man to pass on secret messages to one another, and has evolved continuously to serve our ever increasing demands. 



# CRYPTANALYSIS AND DECRYPTION

We have been breaking codes as long as they have existed. This chapter looks at the science behind code-breaking

**D**igit readers will be familiar with the Crack The Code challenge and the time and effort required for the process of effectively cracking codes and ciphers. This process is called cryptanalysis, which is basically the science of analyzing information systems (commonly called codes and ciphers) in order to study the data hidden inside the systems. Cryptanalysis is used to beat cryptographic security systems, whether they are the pen-and-paper ones devised by you when you were kids, or the latest encrypted security systems anywhere in the world.

## How does Cryptanalysis work?

There are four basic steps to solving any common cryptography puzzle. Although you don't need to stick to these steps strictly, they form a very useful guide for most beginners:

1. **Determine the language being used:** To decode the ciphertext of a coded message into plaintext, you should first have a general idea of what the plaintext is supposed to be like. So, it is important to first identify the language of the plaintext before attempting to look for it. That being said, there's not much to this step except your common sense. If you're decoding your friend's secrets, use the language he/she uses. If you're decoding a top-secret message from France, use French. Pretty simple.
2. **Identify the Encryption being used:** Certain Encryption systems are easily identifiable through the use of some telltale signs, while some others can be quite complex. However, irrespective of the complexity of the system used, once you have identified it, you have taken the first steps towards breaking it.
3. **Find the Key:** Most ciphers and codes use a certain 'key' that helps unlock them. Depending on the complexity of the Encryption system used, this process can be quite painstaking and laborious, but is often absolutely essential. Without a key, most cryptanalysis is simply reduced to brute-force guesswork and trial-and-error.
4. **Decode the Message:** Once you have the key and the ciphertext, you can decode the latter into plaintext. This is usually not the ultimate target of most cryptanalysts, who are simply interested in "cracking the code", unless of course the final message leads to buried treasure or something of the kind.

Most importantly, in the words of a Captain Parker Hitt from a U.S. Army cryptography textbook, "Success in dealing with unknown ciphers

is measured by these four things in order named: perseverance, careful methods of analysis, intuition, [and] luck.”

## Common Cryptanalysis techniques

Coding “secret” information so that it is accessible only to the people you want it visible to is a practice that dates back centuries. And as long as there have been codes, there have been people looking to break them, for reasons ranging from mere academic interest to stopping potential global Armageddon (Enigma and World War II ring any bells?). As is a natural consequence, over the years many different and varied methods of cryptanalysis have been invented and discovered. Most of the current in-vogue methods for breaking modern cryptosystems rely heavily on the use of pure mathematics to solve problems, making them quite complex, and hence beyond the purview of this brief guide.

Instead, we will look at some of the more common cryptanalysis methods that can be used to crack at least the most basic codes. NOTE: These decryption methods are very elementary, and unlikely to be able to crack any of the codes commonly used today. So, restrict using these methods to breaking the code used by your younger brother/sister and watching their astonished expressions as you reveal the information they thought safe from your prying eyes. DISCLAIMER: This writer does not take any responsibility for sibling rivalries/angst created from the abovementioned incident. Use these methods at your own risk!

## Frequency distributions

One of the most elementary forms of cryptanalysis uses something called “frequency distribution”. Essentially, this is a tendency of language where certain characteristics of a language stand out noticeably, in ciphertext as well as plaintext. Someone who has knowledge of these characteristics can use them to break a code in that language quite easily.

For example, in the English language, the letter ‘E’ is by far the most used letter of the alphabet. This means that its occurrence in any message text is mostly the highest. Therefore, when we see a letter being repeated very often, it is safe to assume that this is a replacement of the letter ‘E’. Using this as the starting point, we can discover other letter substitutions and accordingly “crack the code”. Similarly, we can use the common occurrence of the digraphic “th” as a starting point for solving most common codes.

You can use the following table as general guidance for the average percentage of frequency of letters during common English usage:

e: 12.7	t: 9.1	a: 8.2	o: 7.5	i: 7.0	n: 6.9	s: 6.3	h: 6.1	r: 6.0
d: 4.2	l: 4.0	c: 2.8	u: 2.8	m: 2.4	w: 2.4	f: 2.2	g: 2.0	y: 2.0
p: 1.9	b: 1.5	v: 1.0	k: 0.8	j: 0.2	x: 0.2	q: 0.1	z: 0.1	

## Transposition systems

Transposition systems are fundamentally different from substitution systems. In substitution systems, plaintext values are replaced with other values. In transposition systems, plaintext values are rearranged without otherwise changing them. This changes the approach you must adopt when attempting to decrypt these codes.

If you believe that the code you're trying to decrypt uses a transposition system for encryption, arrange the letters in the form of a grid. By trying different variations of number of rows against number of columns, as well as reading off the letters in varying fashion (horizontally or vertically), one will eventually be able to decode the message into plaintext. However, this method can be quite laborious and painstaking, depending on the size of the message to be decoded and the number of letters/variables involved.

## Catch phrases

Frequently, many messages are passed off in code as common language. A notable example of this was the British Broadcasting Corporation's overseas service's use of "personal messages" as part of its regular broadcast schedule. The seemingly nonsensical messages read out by announcers were actually one time codes intended for Special Operations Executive agents operating behind enemy lines. Using such means, the French Resistance was instructed to start sabotaging rail and other transport links the night before the D-Day Invasion of Normandy.

The problem with cryptanalysis of these codes is that they are heavily reliant on the cryptanalyst's knowledge. If the cryptanalyst is unaware of the words to watch out for, the 'key' that indicates it is a coded message, he/she will have to attempt catching the code simply by intuition. Again, the lack of a key will force the cryptanalyst to use brute-force methods to try and decrypt the code, rendering it an extremely time-consuming and prohibitive process. Common catch phrases are: salutations, "please find enclosed", etc which are typical in official communication.

## Syllabary spelling

One of the common keys to breaking into some codes and ciphers is identifying and exploiting syllabary spelling. This essentially includes identification of instances where the same word is spelled in different ways by combining the syllables and letters in different combinations each time. This method is fairly easy if applied to the appropriate code, since most syllabic and letter clusters tend to stick together. By identifying these repeating patterns, one can write down the message in syllabary language, rearrange it on the basis of the sounds of the syllables and the letters and then decode the plaintext message from the rearranged syllabary message. For example, the last letters in words ending with the ‘-ur’ sound are always ‘re’, such as fracture, departure, capture, etc. We can use similar rules that apply to pronunciation of the English language to easily decrypt ciphertext.


## Cryptanalysis today

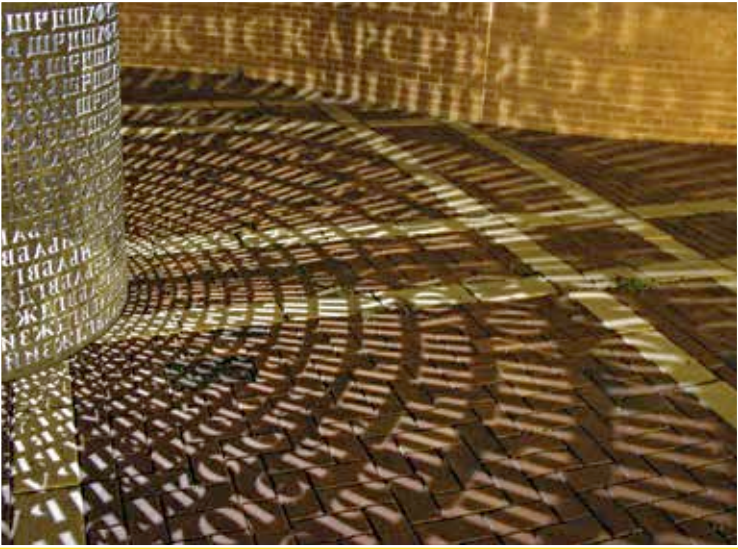
As mentioned earlier, the abovementioned methods are elementary at best, and are unlikely to be of any use in the “real world” of cryptography today. Despite the successful use of computation to break cryptographic systems during and since World War II, the improvement in technology and knowledge has also vastly improved the complexity of new methods of cryptography. On the whole, modern cryptography has become much more resistant to cryptanalysis than the systems of the past.

However, cryptanalysis is far from dead. It has simply had to evolve. Traditional methods of cryptanalysis have given way to new techniques including interception, bugging, side channel attacks, quantum computers, etc. The effectiveness of the cryptanalysis techniques used by government and law enforcement agencies is a tightly guarded secret; but there have been some major breakthroughs recently against both academic (purely theoretical) as well as practical cryptographic systems. Thus, while modern ciphers and codes may be far advanced as compared to even the best of the codes from the past, such as the Enigma Cipher from World War II, Cryptanalysis remains very much active and thriving.

## Conclusion

Cryptanalysis is one of the fields that appears to be shrouded in mystery. Commonly associated with spies and the stuff of detective novels, it is in fact a purely logical field that relies frequently on little more than common

sense, intuition and mathematics. Modern cryptanalysis might be a far cry from the simplistic methods described in this article, yet they are the foundations upon which the field rests. For mathematicians, computer scientists and code-cracking enthusiasts in general, there exists a field much larger than the limitations of what has been explained here very briefly. Happy cracking! 



# THE LORE OF UNBREAKABLE CIPHERS

From time immemorial, mankind has been obsessed with puzzles. The unquenchable thirst to find answers has become the defining characteristic of our species. One could even go as far as saying that this quest for answers has shaped the advancement we see around us

With such an obsession around solving puzzles, it's obvious that unsolved puzzles are by far the most popular, or the ones that capture our imagination. Over the years, there have been many such unsolved puzzles which have caught the imagination of the masses. The notion of invincibility of a cipher has led millions to try their hands at solving them. Despite tremendous advances in the science of ciphers, and using high-end and complex decryption techniques, there are several ciphers still waiting to be translated. In the modern world, where information is power, the field of data encryption has acquired a mighty status. Here we look at some of the famous unbreakable ciphers that have taunted us for years.

## One-time Pad

One-time pad, sometimes known as the perfect cipher, is an encryption algorithm which combines plaintext with a random key to generate a cipher. Provided that the usage is correct, this type of encryption is the only proven unbreakable encryption. Invented in 1917, the one-time pad was derived from the Vernam cipher, which was named in honor of one of its inventors, Gilbert Vernam. The Vernam cipher used a tape for encryption which was looped after each cycle and reused thereby increasing its vulnerability. Owing to its usage by Special Ops teams in World War, intelligence agencies and spies during the Cold War and also in protecting diplomatic communication, the one-time pad gained a reputation for being a simple, yet effective encryption system.



One-time Pad

Each character from the plaintext is encrypted by a modular addition with a character from a secret random key of the same length as the plain text, resulting in a cipher text. The cipher text is impossible to decrypt provided that the key is random, with same or larger length than the plaintext, never reused again and kept secret (obviously).

The pad part of the name of this encryption technique comes from the fact that early implementations involved a pad of paper where the top sheet could be torn off and destroyed after use. Subsequently, huge reductions in the size of the pad was possible. KGB went ahead and created pads that fit in the palm of the hand, or even in a walnut shell. In fact, in a very mission impossible-ish turn of events, highly inflammable sheets of nitrocellulose were once used as one-time pads.

Despite being brilliant in theory, there are certain practical issues with the usage of one-time pads. First and foremost, perfect randomness is required in the one-time pads, which can't exactly be obtained from run of the mill software. High-quality random number generation is difficult to achieve. The random number generation function in most programming languages is not fit for cryptographic use. True randomness is expected to the extent that even if the generating process of a sequence were known up to the last point, it would still be impossible to predict the next event. This simply rules out most running software on computers, which use deterministic processes for random number generation. The next event in a computer is actually pretty predictable, hence they're unusable.

Radioactive decay is an example of a non-deterministic event. Secure generation and exchange of one-time pad material, which must be as long as the plaintext, is another headache. Most importantly, ensuring its proper disposal to avoid reuse and keeping it secret from any adversary is also an issue.

The pad must be kept secure and has to be as long as (or longer than) the message. Also, you wouldn't want to send short messages with a one-time pad, and thus a secure long pad can be used to send numerous messages until you run out of that particular pad cipher.

These practical considerations introduce vulnerabilities which hamper real-world application. Nowadays, one-time pads are rarely used. Implementation difficulties have led to the one-time pad being broken time and again, with breaches serious enough to discourage its adoption as a wide-spread information security tool.

Since the pad is too long for any human to remember, you have to carry them in storage media such as pen drives, DVDs, etc. This is cumbersome, especially when you compare it to modern public key crypto-systems. The risk of compromise during the transportation of the pad, such as someone stealing, copying and returning the pad also cannot be ignored. The effort needed to manage one-time pads for large networks is also a scalability issue.

A famous example of a slip-up using one-time pads: The Soviet intelligence decided to reuse older one-time pads (after many years), as someone assumed this would be pretty secure. This was used to communicate with undercover agents in the UK. British Intelligence picked up on these messages, and noticed a pattern that matched messages they had intercepted many years ago. Needless to say, a large number of encrypted communications were eventually compromised, and there's no doubt a few people found a new permanent home in Siberia, if not worse.

Despite the drawbacks, the one-time pad is still the preferred encryption technique when the encryption work is done by hand. This was the major reason it was so popular in the pre-computer era.

## The Voynich Manuscript

Considered to be the world's most mysterious manuscript, the Voynich Manuscript is a work which dates back to the 15<sup>th</sup> century. The manuscript, which is heavily illustrated with drawings of astronomical events and botany, was bought by a books dealer called Wilfrid M. Voynich, who later took it to the United States of America to have its text deciphered. Almost 100 years later, we still have no clue as to what the text in this manuscript stands for.



The mysterious Voynich Manuscript

The fluency of writing observed in the manuscript indicates that the symbols were not enciphered. There is no delay between characters which would be expected in a written encoded text. However, this could also have been achieved by copying prepared coded text from another source. Further, statistical analysis of the text reveals patterns close to that of natural languages. On the other hand, the manuscript's language is not very different from any other European language in several aspects. Words comprised of more than ten characters are rare and so are words having one or two characters. The distribution of characters is striking as well, because some consistently appear at the beginning, middle or end of the word. The text seems to be more repetitive than typical European languages though, and there are times the same word appears more than twice, consecutively.

Some have suspected Voynich of forging the manuscript himself. As an antique book dealer, he had the requisite knowledge, and a lost book would have been worth a fortune. However, carbon dating of the manuscript put an end to that theory.

The idea that this manuscript is a hoax is supported by the strange features of the text in the manuscript, the suspicious nature of its contents, and lack of any historical proof. It was asserted that the reason no one was able to extract the meaning from this manuscript could possibly be that none existed in the first place. The argument against this suggests that the manuscript is too sophisticated to be a hoax. Hoaxes of that period generally tended to be crude. A simple forgery need not be as intricate as this manuscript, which has many fine touches – some of which are visible only by using more modern tools.

This manuscript has been the centre of attention for many cryptographers across the world, including code breakers from both World War I and II. As of now, no one has been able to solve it and it has become an important part of the history of cryptology. The manuscript was donated to the Yale University where it is known as MS 408.

## The Zodiac Killer

The Zodiac Killer was a serial killer who operated in California, USA in the late 1960s and early 1970s. It is one of the greatest unsolved crimes



Zodiac killer left us all puzzled

to date. Four men and three women between the ages of 16 and 29 were murdered. The police investigated over 2,500 potential suspects, but the forensic techniques of the day were not advanced enough to conclusively convict anyone. The killer identified himself as the “Zodiac” in a series of taunting letters which he sent to the local press. These letters included four cryptograms or ciphers. Until now, only one of the four has been solved. In 1969, one of the four crypto-

grams was cracked by Donald Harden. It contained a misspelled message in which the killer claimed that he was collecting slaves for the afterlife.

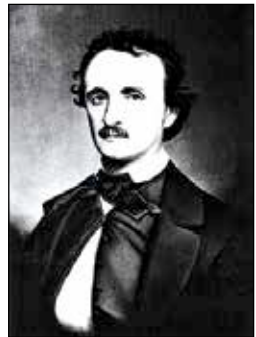
The Zodiac Killer's 'Three part cipher' consisted of three different parts being mailed to three different newspapers in mid 1969. The cover letters were similar and included confessions to earlier killings and threats to kill even more people if his ciphers were not published on the front page of respective newspapers. Further, they also stated that the ciphers contained his identity.

Another one of Zodiac's ciphers was sent to the San Francisco chronicle. It was a 148-character cipher, written at the top of a letter. The cipher was followed by Zodiac killer's crosshair symbol.

## Edgar Allan Poe

Edgar Allan Poe was an American author and cryptography enthusiast. Considered the inventor of detective fiction genre, he pursued activities in the field of cryptography as well. He invited submissions of ciphers and puzzles through advertisements in newspapers and then proceeded to solve them. He went on to publish an essay on cryptography called "A Few Words on Secret Writing". His novels incorporated ciphers and encryption as a part of the story (Yes, just like Dan Brown). His eventual success in the field of cryptography can be attributed less to his knowledge of the subject and more to his knowledge of the magazine and the newspaper culture. Largely ignorant about the field of cryptography, the general public was awestruck by Poe's method of solving a simple substitution cryptogram. Poe created a sensation through his articles and ended up popularizing ciphers in print media, a trend which continues to this day. Many future notable figures in cryptography were heavily inspired and influenced by Poe's work.

Poe released two ciphers as a challenge for his readers. They remained unsolved for a long time, till the first of these was cracked in 1992. A contest was established for the solution of the second cipher which was eventually cracked in 2000. There has been a revival of interest in his work due to the recent movie "The Raven".



Edgar Allan Poe loved cryptography

## The Cyrillic Projector

This is a sculpture in the University of North Carolina which was created by Jim Sanborn, an American artist, in the 1990s. It is an encrypted sculpture and is one of three puzzle sculptures made by Sanborn. The text on the sculpture is in Russian. It is essentially a riddle wrapped on a cylinder. One half of the cylinder is a 16<sup>th</sup> century decoding chart. The chart can be used to decode the other half of the cylinder, which contains a coded text describing the dangers of suppression of intellectual and artistic freedom.



The Cyrillic Projector

This was finally decrypted in 2003 by ElonkaDunin. The sculpture consists of two messages. One is Russian text that shows the usage of psychological control to develop and maintain sources of information. The second message pertains to the scientist Sakharov who was a Nobel Peace Prize winner as well as a Soviet dissident. The message finds its origins in classified KGB files.

## The Oak Island treasure

Oak Island is a 140 acre privately owned island in Canada. It is noted as the location of the “Money Pit” and has been the site of a treasure hunt for over 200 years. Artifacts have reportedly been found as deep as 30 meters, but all such excavations ended in the collapse and flooding of the pit.

In 1975, after observing “strange lights”, a few teenagers discovered a depression on the south eastern end of the island. Upon digging up the depressed area, a layer of flagstones and markings from a pick were discovered. One of the larger stones had an inscription of symbols, and several attempts were made to decipher it. One of the most famous translations is “Forty feet below, two million pounds lie buried.” The pit subsequently flooded.

It has been argued that there is no treasure in the pit, and it is but a natural phenomenon. Such suggestions alleging the pit to be a natural sinkhole date back to early 1900s. There are numerous sinkholes on the mainland near the island, together with underground caves. Appearance of man-made pits has been attributed to the texture of sinkholes. The filling would be softer than surrounding ground leading to the impression that

it were dug up before. The appearance of rotten logs can be attributed to trees falling. Another pit of similar description had been discovered in the surrounding area which lends credence to the natural phenomena theory.

## Kryptos

Kryptos is an encrypted sculpture by Jim Sanborn, who is also the creator of the Cyrillic Projector, located on the grounds of Central Intelligence in Langley, Virginia. The sculpture has been made with four copper plates and other elements such as granite, quartz and wood. There has been speculation about the encrypted message it bears. The name comes from the Greek meaning 'hidden' which signifies the theme of the sculpture, intelligence gathering. The main sculpture consists of four separate enigmatic messages. Three of them have since been cracked. The sculpture was installed in 1990. The first three parts were solved by an NSA analyst in 1992. Notably, a CIA analyst was able to crack the code using pencil and paper techniques. The fourth part is yet to be solved and is one of the most notoriously famous unsolved ciphers in the world. Since 2003, an active Yahoo group coordinates the work of over 2,000 members to decrypt the final sculpture.



Kryptos

Hint for enthusiastic readers: The answer to the first part contains the clue to the final part.

## Vigenère cipher

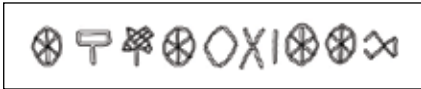
The main problem with simple substitution ciphers is that they are susceptible to frequency analysis. Provided we have a sufficiently large cipher text it can be broken by comparing the frequency of its letters to that of letters in different languages. The aim for cryptographers was therefore to make encryption techniques that were immune to frequency analysis. A common approach is to suppress the frequency data using more than one alphabet to encrypt the data. The Vigenère cipher is one such method which is a simple form of polyalphabetic substitutions.

The method was originally developed in the 16th century but was later misattributed to Blaise de Vigenère and is now known as the Vigenère

cipher. The cipher is very well known because it is easy to understand and implement and appears unbreakable to beginners. This is where it acquired its reputation of being unbreakable. Spurred on by the myth, a number of people have tried to implement the encryption scheme of this cipher, only to have them broken. Charles Babbage was the first person to have broken a variant of this cipher. The main weakness of this cipher is the repeating nature of its key. If the key's length can be correctly guessed, then the cipher text essentially becomes a combination of interwoven ciphers, which individually are easily broken. Efforts were made to improve this cipher, which led to the development of the one-time pad.

## Indus valley script

The term Indus valley script refers to the strings of symbols associated with the Indus valley civilization in use during the middle of the second millennium BC. It is not generally accepted that this script is used to record this language. The first publication of Harrapan seals dates back to the



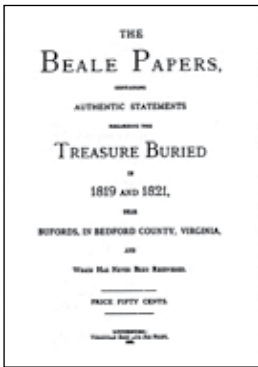
Indus valley script

1870s. A number of objects bearing symbols from the Indus valley script have been discovered in the past few years. It was believed that this script is related to

the Brahmi script. After multiple attempts to decipher the script, there is still no success in sight and it is still listed as undeciphered. The language for which this script was used has also not been identified yet. Over the years, a number of solutions have been proposed, but none have been widely accepted. One of the biggest challenges in the deciphering process is that no language has been identified, although some words from the Rigveda are sometimes used for comparison. The fact that the average length of the inscriptions is quite small only adds to the complexity. Further, no bilingual texts have been found. The topic is extremely popular among amateur researchers and various decipherment claims have been seen, though none of them have found any recognition.

## Beale ciphers

The Beale ciphers are three ciphers which state the location of a buried treasure worth about \$63 million in today's money. Two of the ciphers describe (unessential) details such as the contents of the treasure and



Beale Papers

the names of the people in the family of the original treasure owner. The story originates from an 1885 pamphlet that claimed a buried treasure was hidden in a secret location in Virginia in 1820. All three cipher texts were published in the pamphlet. The cipher detailing the contents of the treasure was deciphered using the Declaration of Independence of the United States as the key. The other two ciphers remain unsolved. Later on there has been considerable debate on whether the cipher texts are real or hoax. It has been alleged that there is evidence based on the kind of words

used, proving that the document could not have been written at the time it says it was.

## Dorabella Cipher

This cipher is an encrypted letter written by Edward Elgar to Miss Dora Penny in the late 19<sup>th</sup> century. Edward Elgar was a music teacher, and Dora was almost seventeen years his junior. The letter could not be decrypted. The cipher consists of 87 characters, is over three lines long and seems to be made up from 24 symbols that consists of up to three semi circles oriented in eight directions. The symbol frequency is similar to that of English, but attempts to decipher it as a simple substitution cipher have not yielded any results. Some theories regarding the solution presume that the solution is not text but a melody and the 8 different positions indicate notes of the scale. In commensuration of Edward Elgar's 150<sup>th</sup> birth anniversary, the Elgar society heavily advertised the cipher.

## Linear A

Linear is one of the two writing systems used in ancient Crete along with Linear B. Primarily, Linear A was used a script for religious purposes. Linear A is generally understood to be used between 1800 to 1450 BC. It was discovered by Sir Arthur Evans in 1900



Linear A

(AD). A large number of clay tablets inscribed with mysterious symbols were found. Sir Evans believed he had discovered the mythical palace of King Minos and the Minotaur's labyrinth, and christened the inscriptions and languages they represented as 'Minoan'. Evans spent his remaining life trying to decipher the inscriptions with limited success. Three writing systems were used, including somekind of hieroglyphic script, Linear A and Linear B. Out of these three only Linear B has been deciphered, the rest are yet to be understood. It is believed that the hieroglyphic script led to Linear B which led to Linear A, though the relationship is unclear. Linear B was largely deciphered in the 1950s. Although Linear A and Linear B share a lot of symbols, Linear A is still a mystery waiting to be solved.

## Phaistos Disc

The Phaistos Disc is a disc of clay obtained from Crete, Greece from about 2000 BC. It is covered on both sides by a spiral of stamped symbols. Its



Phaistos Disc

purpose, meaning and where it was manufactured continues to be a mystery making it one of the most famous archaeological ciphers. The disc was discovered by Italian Luigi Pernier in 1908. It features a number of tokens and unique signs presumably made by pressing of discs into the soft clay. Numerous attempts have been made to crack the cipher. Despite existence of uncertainty on whether there is script behind the signs, most decipherments

assume this. Attempts at deciphering the code seem unlikely to succeed without further examples of signs. It is believed that there is not enough material available for a meaningful analysis. It is also speculated that this disc could be used to decipher Linear A.

## Rongorongo Script of Easter Island

Rongorongo is the system of writing discovered on Easter Island. The people of Easter Island were probably inspired to invent the Rongorongo script after seeing the writing used by the Spanish. Rongorongo was used till the 1860s after which the knowledge was lost. With the advent of missionaries, attempts were made to understand the language. Numerous attempts have been made at decipherment; none have been successful

so far. Some information has been identified as their perception of a calendar, but the majority remains unread. If Rongorongo does indeed prove to be writing, it will be one of the few independent inventions of writing in the history. Assuming Rongorongo is truly writing, there are three serious challenges to cracking the cipher including the small appearances of text, the language on which it is based and no context whatsoever in which to interpret them.



Rongorongo Script

## Bulambod

Bulambod is a type of encryption software which is freely available online. It's a cipher algorithm which is allegedly not based on existing algorithms.




Bulambod

It was not invented recently, but was started in the 1990s with an aim to make an unbreakable cipher. Its main strength is the length of the password it allows. It further allows the user a possibility to use a number system of his own choice for encryption.

While there are a number of ciphers that have not yet been deciphered, their number is declining steadily. Modern computation techniques and improvement in algorithms, are equipping us with better tools for decryption. Of course, better computing machines also mean that we can get better at encryption techniques as well, and it seems for now that the balance is holding up quite well.

## References:

1. [http://www.pro-technix.com/information/crypto/pages/vernam\\_base.html](http://www.pro-technix.com/information/crypto/pages/vernam_base.html)
2. <http://www.cryptomuseum.com/crypto/otp.htm>
3. <https://mice.cs.columbia.edu/getTechreport.php?techreportID=1460>
4. <http://beinecke.library.yale.edu/digitalibrary/voynich.html>
5. <http://www.archive.org/details/TheVoynichManuscript>
6. <http://www.ciphermysteries.com/>

7. Cryptography, Information Theory, and Error-Correction: A Handbook for the 21st Century (<http://books.google.com/books?id=fd2LtVgFzoMC&pg=PA21>)
8. Everyday Cryptography ([http://books.google.com/books?id=1NHli2uzt\\_EC&pg=PT142](http://books.google.com/books?id=1NHli2uzt_EC&pg=PT142))
9. <http://www.cs.trincoll.edu/~crypto/historical/vigenere.html>
10. <http://www.people.ku.edu/~jyounger/LinearA/>
11. <http://themodernantiquarian.com/site/10857/phaistos.html#fieldnotes>
12. <http://www.newscientist.com/article/mg15020344.300-cracking-the-easter-island-code.html>
13. <http://www.und.edu/org/crypto/crypto/general.crypt.info/Cyrillic.cipher>
14. <http://www.baconscipher.com/OakIsland1.html> 



# DIY: MAKE YOUR OWN CIPHER

So now that you know a decent bit about cryptography, how about making your own little cipher?

**O**K, enough theory, let's get down to actually doing something now. In this chapter, we'll guide you through the rather interesting journey of creating your own cipher.

The first question that crops up is "What cipher should we create?"

Classical ciphers are the best examples, and are the easiest ones as well. The problem with them is that they work on the level of ‘letters’. Classical ciphers don’t let you go below that level. In our daily life, we deal with a lot of data that’s not just a collection of letters. For example, there’s no easy way to convert an image or a video to letters. Of course, the word ‘letter’ is being used loosely here. Another question could be “Letters in which language?” Modern encoding techniques can represent letters from quite a long list of languages using clever techniques to organize bits. This leaves us with one option – abandon classical ciphers because classical ciphers cannot be used to encrypt data well enough in a digital world.

Coming to modern ciphers, we have two options: asymmetric algorithms (or public key algorithms) and symmetric algorithms (or private key algorithms). It’s well known that asymmetric algorithms provide more

security by allowing the key used for encryption to be known by anyone. However, most asymmetric algorithms are based on very complicated mathematical calculations. Though we’re going to create our own algorithm, and we would want to make a secure one, designing an asymmetric algorithm would require quite a lot of explanation and in-depth understanding of number theory, which is beyond the scope of this book.

So let’s stick to a symmetric algorithm, and move on to creating it. We’ll do this stepwise.

## NOTE:

For this example that will get you started, you need to remember that this won’t be the world’s best symmetric cryptographic algorithm. The sole aim of this chapter is to show how minute details affect the overall procedure of creation and implementation of algorithms. As we move forward, we’ll show you points where you can change the variable factors of this procedure to alter the resulting algorithm. There are points that may appear to be flaws in the design process, but we’ve put these in deliberately to show you why a well thought-out method must be used, and how unnecessary complications do not necessarily make a cipher useful or strong. Nonetheless, in the end you’ll have an encryption algorithm. This algorithm will be very weak for any practical encryption (unless you only want to fool your family and friends with it).

## Step 1

The first factor to be considered before starting the actual design procedure is to decide on the block and key length. The bigger both are, the better

the security. This is because small blocks and/or small keys can make the ciphertext vulnerable to cryptanalysis attacks.

Of course, you should also know that as the block size and key length increases in length, the cipher process slows down. All CPUs are designed to work with a certain amount of data per cycle. For example, 32-bit computers work with 32 bits per cycle, while 64-bit CPUs can process 64 bits in one cycle (max). So when you make the block size larger than that, each block would need multiple cycles to be transformed and that will slow down the entire process. We'll be using a 32-bit key and a 32-bit block size.

## Step 2

Since we're creating a process which can scramble and unscramble messages using the same key, we obviously need to find a mathematical process that's reversible. Step 2 is choosing that method to serve as one of the key parts of the algorithm. For this example, we're choosing a simple 'XOR' method. The table will illustrate the XOR method:

**TABLE 1: HOW XOR WORKS**

First bit	1	0	1	0
Second bit	0	0	1	1
XOR Result	1	0	0	1

### NOTE:

We're going to obtain the bits on the 'second bit' row from the 32-bit key we've decided but we won't use that directly. The reason we would do that is because it makes the whole algorithm just an XOR operation which is too easy to reverse (using a trial and error method) if the key is known to the other person (we're designing a symmetric algorithm, remember?). We're going to make it a little more complicated.

The idea is simple: You take two bits. If both bits are the same (either both 1s or both 0s) then the result is 0. If they're different, the result is a 1. Notice that if you reverse the order of the rows and read bottom up (treat the last row as the First Bit and the First Bit as the XOR Result row) even then the results hold true. This is the simplest way to know that the process is easily reversible. In our example, the first bit row is the actual data, and the second bit would be the key. As long as you know the key (Second Bit), you can encrypt and decrypt.

## Step 3

As we said earlier, small block-and key sizes are a bad combination that

can be cracked easily. However, that's true only if the "algorithm" is known by the person trying to crack the cipher (someone who's not the intended recipient). In this case we won't be increasing the block size but altering the key size so that the strength of the cipher increases. You could also increase the block size to further increase the strength, but we're trying to keep things simple to understand.

We're going to increase the key size using a very simple technique – we take the original key (32-bit) and split it into four parts. Remember, each bit sequence of 4 bits corresponds to one hexadecimal number. We will use that here instead of dealing with bits directly to make it easy to read. Our key when represented in hexadecimal would be a sequence of 8 hexadecimal numbers. For simplicity, our key will be "02468ACE".

We're going to derive a 128-bit key from the original 32-bit key by working with chunks of 8 bits from the original key using the following table:

TABLE 2: GENERATING THE 128-BIT KEY FROM THE 32-BIT KEY				
	Part 1	Part 2	Part 3	Part 4
0th Iteration (original key)	02	46	8A	CE
1st Iteration	CE	02	46	8A
2nd Iteration	8A	CE	02	46
3rd Iteration	46	8A	CE	02

If you notice, the pattern is actually simple: we rotate the chunk's positions by one and we do this three times. We then write them in sequence making the derived key:

02468ACECE02468A8ACE0246468ACE02

We'll use this key to run the main algorithm. Before we proceed, there are a few things you need to note:

- ▶ The algorithm we've used to derive the 128-bit key from the original 32-bit key is very weak and can't be used for any real secure communication.
- ▶ There are better ways to generate a 128-bit key from a 32-bit key. For example, you can do bit shuffling rather than shuffling blocks of 8-bits. By nesting rotations (rotating bits of chunks, as well as rotating the chunks themselves) the generated sequence would make the procedure of tracking the 32-bit key from the 128-bit key a bit more difficult.
- ▶ We could have chosen a 128-bit key in the very beginning, it would have served as well, but then, we would have to send the 128-bit key – thus making it impractical. We do the 32-bit to 128-bit transformation so that

the algorithm doesn't make use of the key directly but first obtains the 128-bit key.

Now, some of you will be wondering how on earth we're going to encrypt a 32-bit block-size plaintext with a 128-bit key size. Also, if we're using XOR, how do you XOR a 32-bit block with a 128-bit block? That's what we'll show you in the next step.

## Step 4

Since the plaintext block size is 32 bit and the key length is 128 bit, they cannot be involved in a direct XOR operation. So what we do here is take consecutive 32-bit blocks from the plaintext and XOR them with consecutive 32-bit blocks of the key. Following this method, the key would get consumed by the fourth block of the plaintext ( $32 \times 4 = 128$ ). The fifth block of the plaintext would then be XORed with the first 32-bit block of the key, and so on. Hence, if the plaintext is "FDA72BE910CA8AB3C1068EE-A735715BCFF2938A31029ABD37ECA928CE7EA53C3" and the 128-key is "02468ACECE02468A8ACE0246468ACE02" (which we got in Step 2) then the XOR operation would be done with chunks in the following manner.

### NOTE:

There are better ways to get a 128-bit key, given another key is known. One of the best ways is to use hashing techniques. A good example would be the MD5 algorithm. This algorithm takes in a variable sequence of inputs.

**TABLE 3: GETTING THE ACTUAL CIPHERTEXT USING THE PLAINTEXT AND THE 128-BIT KEY**

Part of plaintext	FDA72BE9	10CA8AB3	C1068EEA	735715BC	FF2938A3	1029ABD3	7ECA928C	E7EA53C3
Part of key	02468ACE	CE02468A	8ACE0246	468ACE02	02468ACE	CE02468A	8ACE0246	468ACE02
Resulting ciphertext	FFE1A127	DEC8CC39	4BC88CAC	35DDDBBE	FD6FB26D	DE2BED59	F40490CA	A1609DC1

We're going to concatenate the 32-bit blocks of result into one to get the ciphertext. Hence the resulting ciphertext would be:

FFE1A127DEC8CC394BC88CAC35DDDBBEFD6FB26DDE2BED59F-40490CAA1609DC1

## Decryption

Decryption of the ciphertext is straightforward in this algorithm. Of course, the two things we're going to need before beginning the process are the original 32-bit key and the entire ciphertext. We then are going to derive the 128-bit key from the 32-bit key using the process in Step 3. Once we have

that, we're simply going to do the XOR on each 32-bit block of the ciphertext. It would be something like this:

TABLE 4: DECRYPTING THE CIPHERTEXT WITH THE 128-BIT KEY								
Part of ciphertext	FFE1A127	DEC8CC39	4BC88CAC	35DDDBBE	FD6FB26D	DE2BED59	F40490CA	A1609DC1
Part of key	02468ACE	CE02468A	8ACE0246	468ACE02	02468ACE	CE02468A	8ACE0246	468ACE02
Resulting plaintext	FDA72BE9	10CA8AB3	C1068EEA	735715BC	FF2938A3	1029ABD3	7ECA928C	E7EA53C3

We can then concatenate the resulting blocks of plaintext to arrive at the original plaintext. Simple and easy, but was our design really any good?

No. It wasn't. Not only is the encryption easy to break but the design process contains a perspective which is unnecessarily complicated and can be simplified.

### How our design process is flawed

While the algorithm does look real good, there is a flaw in its design. The question is where? Let us reconsider the way we're doing the XOR operation. It's in a sequence of blocks. The first 32-bit block of plaintext and the first 32-bit block of the key are being XORed. The second 32-bit block of plaintext and the second 32-bit lock of the key are being XORed. Similar for the 3rd and 4th blocks of plaintext and the key. For the fifth 32-bit block of plaintext, we're reusing the first 32-bit block of the key. If you pay just a little attention, all that we're doing is to XOR the first 128 bits of the plaintext with the full 128-bit key and the next 128 bits of plaintext with the key again. The table we used to show the conversion of plaintext to ciphertext (table 3) could also have been shown as in the following table (table 5).

TABLE 5: SIMPLIFIED VERSION OF ENCRYPTION PROCESS USING 128-BIT BLOCKS OF PLAINTEXT AND COMPLETE 128-BIT KEY (A REFACTOR OF TABLE 3)		
Part of plaintext	FDA72BE910CA8AB3C1068EEA735715BC	FF2938A31029ABD37ECA928CE7EA53C3
Part of key	02468ACECE02468A8ACE0246468ACE02	02468ACECE02468A8ACE0246468ACE02
Resulting ciphertext	FFE1A127DEC8CC394BC88CAC35DDDBBE	FD6FB26DDE2BED59F40490CAA1609DC1

By such behavior we can say that the block size that the algorithm is actually using is not 32 bits but 128 bits. Instead of thinking that we're running the algorithm in chunks of 32 bits, you can rather think that we're

operating in chunks of 128 bits instead. By changing the perspective, we would actually be simplifying the algorithm. But then again, the simplification would only be in terms of thinking, not in terms of writing it as a program for the computer.

## A view from the implementation perspective

We said earlier that implementations of any algorithm on a machine are limited by the capabilities of the machine and that 32-bit machines can take in only 32 bits and 64-bit machines can take only up to 64-bits in one cycle. Hence, when implementing this algorithm on the machine, you cannot just stuff up all the 128-bits of the plaintext in a variable and 128-bits of the key into another variable and run the XOR operation. The XOR operation is always done on bits and the number of bits that a machine can take in is limited. Hence, your implementation as such will have to consider those boundaries. In such a context, you will have to go with either chunks of 32 bits or 64 bits depending on the machine on which you're going to run the encryption.

It's the same reason why we have different setup files for 32-bit and 64-bit architectures for the same version of the same software. The original perspective of running the XOR operation in blocks of 32-bit was not all that bad. After all that's how you would be doing it on a 32-bit machine.

## Suitable programming languages for implementation

Just because you can create beautiful web pages with great functionality using PHP and Javascript does not mean you can use those for implementing the cipher we just designed. Some factors which would govern the decision on the programming language to be used for implementation are:

1. **Features provided by the language:** While PHP and Javascript can be used to do bit-level operations, those are not the things these languages were designed for. The use of PHP and Javascript remains on a higher level than playing with bits. A little lower level programming language such as Java, C, C++, etc., would be what you need.
2. **Speed:** PHP and Javascript are not really programming languages in the classical sense. These are scripting languages. Though you do write programs in them and they do work, the programs are not compiled. They are interpreted. While Java has a rich range of functionality and can work on bit-levels, it is not really fast at doing that (after all, Java programs

too are bytecodes which are interpreted by the JRE). A programming language which when compiled creates high performance code which can run on multiple architectures is deemed ideal for such usage. While Java is cross platform compatible, it's not faster than C/C++ for bit-level operations. Thus C/C++ are thought to be the best languages to write high performance ciphers with complicated mathematical procedures. As a side note, we would like to say that some people argue that Haskell is high performance as well, but benchmarks show that it's not faster than C/C++ (implementation of Haskell itself happens to be in C/C++) and thus it does not really outperform C/C++.

Since most modern ciphers deal with mathematical calculations, C/C++ would qualify easily as the duo comes closest to hardware while maintaining high speed for calculations.

## A missing piece

We told you earlier that we were not going to illustrate the best ciphering technique. However the algorithm that we have designed has a missing piece in it: the cipher will fail for any data whose size is not a direct multiple of 128 bits (16 bytes). There are multiple ways to do just that. One of those ways would be to pad the plaintext with random data to make its size a perfect multiple of 128 bits and send along the size of data that's actually to be considered after decrypting along with the key.

## Disadvantages of using your own cipher


Nothing comes without its problems. A custom-designed cipher such as the one we illustrated in this chapter are no exceptions.

1. The first disadvantage of making your own cipher is that it's going to be weak (unless you're a mathematical whizkid). Well established ciphers are the result of years (if not decades) of mathematical research. It's next to impossible to design a new algorithm overnight that would be cleverer than years of research done by actual mathematics whizzes.
2. The second disadvantage is that no one except you would be able to use it for practical purposes. The software to encrypt and decrypt data using well known algorithms is usually already installed in the operating system or other software which is needed to use them (e.g. web browsers). Your algorithm is going to be just yours. You know the design and how it's implemented, and no one else can make use of it. Not only that, you need to transfer the algorithm to everyone who wants to use it.

3. You need to implement the cipher yourself. By that we mean to say that if you're going to use your own cipher on your computer, you need to write the program yourself as well. This can take time and can be imperfect or make it insecure if the person who is writing the algorithm skips a step.
4. If you encrypt something using your own cipher and then forget the algorithm, or lose the program you use to encrypt and decrypt data, you're literally locked out of your own data. No other program on this planet is going to help you there. Make sure you have saved the algorithm and the program's source code somewhere safe other than your own computer (cloud services are a good choice), so that if you ever need it back, you can have it.

## Advantages of using your own cipher

Reality is that no one wants to do anything unless it has got at least some advantages. So what are the advantages to having your own cipher? The first advantage (and the only big one) is that it's yours. You know how it works and you alone know it. So if you're going to encrypt something using your own cipher and send the data (or store it), you alone will know how to decrypt it! For a cipher to be cracked, the key needs to be known; but that's not the only thing needed. You also need to know the algorithm. If someone gets the key but does not know the algorithm, you're still relatively safe.

You can implement your own cipher as a program which can be used to 'password protect' data – the password you enter should act as the key which can then decrypt the data. If the wrong key is supplied, the decrypted result would be garbage. As you can see, this technique can be used with only what you use; after all it's your own cipher and you would have wanted to use it only for yourself. Of course you can give the cipher and the program you created to your friend, but the cipher would not be accepted as IETF for being utilized in [HTTPS](#) anyway and it would remain private. 



# THE FUTURE OF CIPHERS

Change is the only constant, because nothing else can afford to stop

## What Ciphers Used To Be

When one day, a man sits down to analyze everything that computers have affected (and by default, improved) in the existence of the human race, that man would be doomed - that work would never get done completely. We're saving ourselves and you a lot of time by not going into the details of the issue, but lets just say that computers are awesome, and leave it at that. So, ciphers. To give you an idea of what was 'groundbreaking' in ancient times, try cracking this.

L O R Y B G L J L W.

We would've loved to see your chain of thought here, but let's get to the point here. That meant:

I LOVE DIGIT.

Look closely, and you'll see that all we've done is shift the alphabets by 3. You might now remember this from the general quizzes to look as a child.

Now this exact code is said to have a great history - people claim that Caesar himself devised this, and used it to keep his political adversaries from understanding his messages across the empire, even if they got hold of the message. Now, this code may have proved effective then, but this code has scarcely fooled anyone since. But it continued to be used, because the main area of application of cryptology in olden times was the battlefield, and in devising a new code, many areas are to be considered, not least the intelligence of the man who sits to write down the encoded message. Too complex, and the sender might mess up the sending, or the receiver might mess up understanding. Enter, the computer.

## What Ciphers Are Today

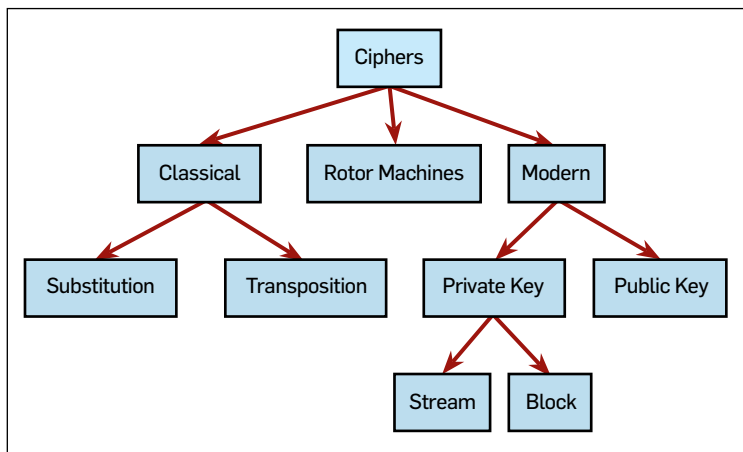
If Caesar was a proud man, he would feel incredibly belittled by the heights that a field that he was one of the chief propagators of, has reached. Where we stand today, the kind of algorithms that are used to secure our world would take the brute force attacks anywhere between  $10^{10}$ - $10^{20}$  years to crack. Did we mention that it would take the parallel processing power of all the computers in the world, along with mobile, laptops, tablets, and everything else that says  $1+1=2$ .

Yeah, it all seem well and good for as long as we talk in such grand scale of things. There is one problem, though. And it is a huge problem. It is...

## Cryptography in a Quantum World

Ah, well. Finally we get to discuss this issue in some detail. Last month, we gave you a teaser of the capabilities of Quantum sciences in the 'How Safe is Your Online Transaction?' chapter in the Fast Track to





Different kinds of ciphers out there

E-commerce. Now is the time we get down to what exactly prompted us to speculate of such unpredictable times.

So, at the last page, we left the issue of computers at ‘computers are awesome’. Well, they are. But they’re not as awesome as they can be. Not yet.

Let us treat you with an example here.

We give you a number. Say, 287. Now, we tell you to factorize it. Those of you who do not get chills solving class 5th problem at this point (need not worry if you are in class 5th right now, though), might come up with something like  $2^{10} + 3^3$ . Good on you, sir. What you did was absolutely incredible, if we’re talking in terms of computing power. This is called the factorization problem, and this (though not in such an easy form, obviously) is what is used in all forms of cryptography known to you. Because, as it turns out, our computers cannot factorize that well. In the grand scale of things, they cannot do it at all. And that’s why RSA loves this problem - so much so that their entire work model is structured around this problem. That is the reason the world fears Quantum Computing as much as it lusts for it. The moment someone announces that a full fledged quantum computer has been built, panic would ensue across everything that is based on money and the internet. So basically the global economy.

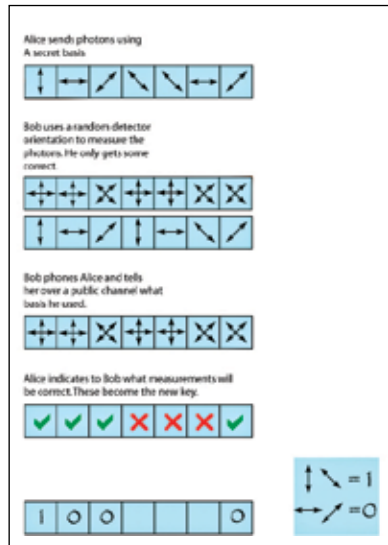
Why, you ask? Well, as it turns out, a computer that can use qubits (quantum bits) instead of bits can crack this problem in about as much time as it took you to read the last word of this sentence. Interested (and really

really smart)? Check out Shor's algorithm for prime factoring. Really, please do. There are few mathematical formulae in the world that can bring it to its knees the moment one knows how to use them. No kidding.

## Quantum Cryptography

We might not have built a quantum computer, but the field has seen incredible advancements, and today you can possibly use quantum techniques in your friendly neighbourhood ciphers.

Let us take, for example, something called the BB84 cryptography protocol. The super stripped-down idea is that if Alice wants to send data to Bob (because apparently, when it comes to a quantum situation needing 2 people, it is always Alice and Bob), what would be done is that Alice would encode her data in qubits (something like photons - the light particles), and send them to Bob in some arbitrary bases. Now, Bob would receive the inflow in some arbitrary bases of his choice. His results would be phased by the difference of bases chosen by him and those chosen by Alice. This problem, though can easily be corrected by a simple communication to clear the air on this topic. Oh, and forget about eavesdropping, because if a third listener tried to copy the data with them, it would not be allowed. The network won't stop him, and neither would Bob or Alice. That's because they don't need to - the Universe is on it already. It is impossible to copy quantum bits exactly.



The process explained graphically

## Post-Quantum Cryptography

There are many, many allegations that you might throw at mankind, and most of them would probably be true. But one thing that you can never say that humans lack is vision. Yes, today's attempts at Quantum Com-



puting are a far cry from the kind that would be needed in order to tackle the awesome power bursting from the proper concept that is Quantum Computing. But because no concept would ever be allowed to run loose in today's world, counter-ideas for the force of Quantum Computing are topics of hot debate, and of course, research.

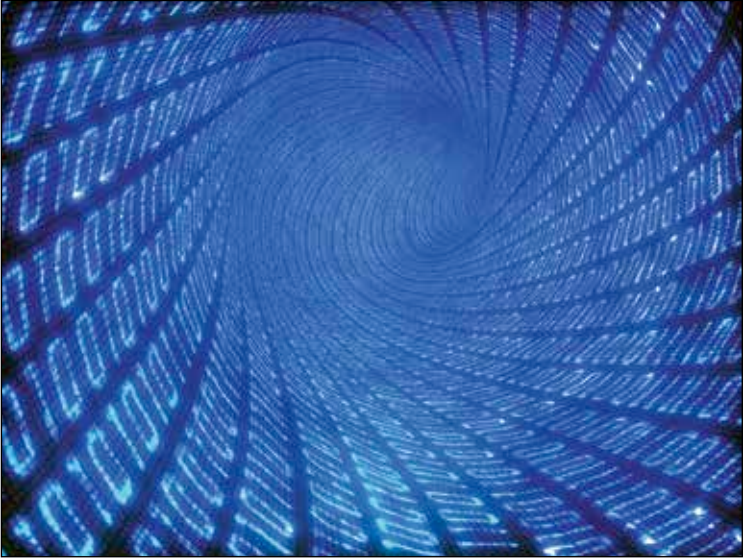
It is well known that unless we stumble upon another hitherto unforeseen blockage in nature (we're looking at you Heisenberg Principle), Quantum Computers would make it through. That day might not be today, and it might not be

tomorrow, but it will be here someday. And if a power exists, we need to consider that it will be used for at least some sort of evil, especially when it holds such never-seen-before capabilities. Consider this: the moment the day comes that a Quantum Computer is announced to the world, RSA is dead, DSA is dead; hyperbolic, elliptic, hyperelliptic, and every other curve based encryption technique is dead. And if we really want to make sure that every form of data protection technique known to man is not dead with all of that, we need to find remedies. And we need to find them now.

There are some ideas in that space in existence, though. These include:

### 1. Hash based cryptography

If you have any background in computer science, you might have heard of hash tables. And if you have some serious background, you probably love them, like some of us. To hear that these little things are quantum proof in the broad sense is a huge relief. Yes, folks, hashing would survive even in the era of quantum computers, because the idea here is that we shouldn't be solely reliant on a secret trapdoor functions to produce our keys for us. Then again, if random number based keys are the only hope for the future, then it might not be that great after all.




## 2. Code based cryptography

Wild Goppa codes are codes over small finite fields  $GF(q)$  obtained from Goppa polynomials (yes, that name). While the whole process of polynomial based cryptography is as old as our regular, everyday public key systems, they are rarely used due to the unreal key lengths they produce, in comparison to the much loved (and feared) RSA.

## 3. Lattice based cryptography

As you might have guessed, this kind of cryptography is based on lattice study. And there is no easy way to say this, but a lattice  $L$  is the set of points in an  $n$ -dimensional Euclidean space,  $R_n$ , each with unique basis. If you must know, a basis of  $L$  is a bunch of vectors, uniquely represented as a set of numerical coefficients.

If you find these terms to be challenging/interesting/overwhelming/all of these, we suggest that you look it up yourself, because not only is it beyond the scope of the topic, but also this chapter, and the FastTrack itself. But, it is interesting, nevertheless. We, here, spent quite a few afternoons going through this incredible pool of knowledge ourselves. 



[illegible]





All this and more in the  
world of Technology

**VISIT  
NOW**



[www.thinkdigit.com](http://www.thinkdigit.com)

# Join 70000+ members of the Digit community



<http://www.facebook.com/thinkdigit>

**facebook**

**digit**  
magazine

**Like**

Your favourite magazine on your social network. Interact with thousands of fellow Digit readers.

**Like**

Wall  
Info  
About  
Contact Us  
Subscribe  
MP3s  
Channel  
Photos  
Notes

<http://www.facebook.com/IThinkGadgets>

**facebook**

**I Think Gadgets**  
Corporate/Institution

**Like**

An active community for those of you who love mobiles, laptops, cameras and other gadgets. Learn and share more on technology.

**Like**

Wall  
Info  
Photos  
Videos  
Events  
Questions

<http://www.facebook.com/GladtoBeAProgrammer>

**facebook**

**Glad to be a Programmer**  
Corporate/Institution

**Like**

If you enjoy writing code, this community is for you. Be a part and find your way through development.

**Like**

Wall  
Info  
Photos  
Videos  
Notes  
Events

<http://www.facebook.com/devworx.in>

**facebook**

**devworx**  
Individual

**Like**

devworx, a niche community for software developers in India, is supported by 9.9 Media, publishers of Digit

**Like**

Wall  
Info  
Friend Activity  
Questions  
Photos  
Welcome  
Videos